ADAPTIVE FILTERING AND PATTERN RECOGNITION OF EVOKED POTENTIALS

By

GOPAL POOVANPILLI MADHAVAN

A thesis

Submitted to the School of Graduate Studies

in Partial Fulfilment of the Requirements

for the Degree

Doctor of Philosophy

McMaster University

 November 1985

ADAPTIVE FILTERING AND PATTERN RECOGNITION

OF EVOKED POTENTIALS

DOCTOR OF PHILOSOPHY (1985) McMASTER UNIVERSITY

(Electrical and Computer Engineering) Hamilton, Ontario

TITLE: Adaptive Filtering and Pattern Recognition of Evoked Potentials

AUTHOR: Gopal Poovanpilli Madhavan,
B.Sc. Eng. (Calicut University)
M.S. (Indian Institute of Technology, Madras)

SUPERVISOR: Dr. H. de Bruin

NUMBER OF PAGES: xii, 147

ABSTRACT

The problem of estimating evoked potentials and its pattern recognition and classification is addressed in this thesis. After providing the relevant physiological background and reviewing the various methods of processing the evoked potential, we propose the method of adaptive noise cancellation for estimating the evoked potential without stimulus repetition. A new weighted exact least squares lattice algorithm is derived for this purpose. The variable weighting factor can be used to make the algorithm robust. Its performance is compared to that of unnormalized and normalized exact least squares lattice algorithms and is shown to be superior. One example of using adaptive noise cancellation to estimate evoked potential without stimulus repetition is presented. Pattern recognition of evoked potentials is achieved by syntactic methods. We derive a finite-state grammar to represent the normal evoked potential. Suitable preprocessing using a zero-phase bandpass filter, parsing and attribute checking are the steps in this classification procedure. A database of normal evoked potentials and optimized acceptance criterion are used for checking the attributes. Detailed training and test runs are performed to demonstrate the performance of this classifier.

DEDICATED TO

the loving memory of my father

and to my mother

ACKNOWLEDGEMENTS

It is with deep appreciation that I wish to acknowledge the considerable efforts of my supervisor, Dr. H. de Bruin, aiding both in the performance of this research and in the preparation of this thesis. His guidance was immeasurably valuable and without it, this work would not have been as fulfilling.

I would also like to thank the other members of my supervisory committee, Dr. N.K. Sinha and Dr. A.R.M. Upton for their guidance, constant support and encouragement. I am indebted to Dr. A.R.M. Upton for allowing me to pursue my own interests and for my present job.

Dr. S. Haykin and Dr. M.E. Jernigan of Waterloo University deserve my special thanks for introducing me to adaptive filtering and pattern recognition. Their continued support and enthusiasm for my work have been invaluable. I would like to thank Dr. M.E. Towell for giving me an opportunity to branch out and do fulfilling work for the past three years and for the financial support.

I am indebted to B.W. Dahanayeke and J.R. Raol for the many interesting discussions and the continued help they have provided me.

Without being exhaustive, my other colleagues, S. Puthenpura, G. Jasechko, D. Bhattacharya, R. Epstein, V. Epstein, J. Johnson, H. Kaur and P. Shaw deserve my special thanks for their friendship and support.

Appreciation is expressed to NSERC and McMaster University for financial support. I am also grateful to J. Arsenault, B. Petro and L. Honda of the Wordprocessing Center for their cheerful and efficient typing of parts of this thesis.

Finally, it is with deep gratitude that I acknowledge the help my family has provided me in all my endeavours. Without their constant encouragement, none of this would have been possible.

CONTENTS

PART B - ADAPTIVE FILTERING

PART C - PATTERN RECOGNITION

LIST OF ILLUSTRATIONS

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Bioelectric activity within the human body, when properly acquired and interpreted, is an enormous source of information about the dynamic aspects of body functions. Electrical activity of the large body of neurons in the brain gives rise to measurable potentials on the surface of the scalp. In the absence of any specific stimulation, the measurable electrical potential on the scalp is called the **'electroencephalogram' or e.e.g.**. This can be considered to be spatial and temporal summation of the non-specific electrical activity of neurons in the brain.

When a discrete amount of energy impinges on sense organs, only a specific population of neurons responsible for that sensory modality is activated. The electrical activity of the specific neuronal population that was activated can be identified, knowing the time relationship between the stimulus and the electrical response that it generated. These responses are called **'evoked potentials' or EPs.**

1.1 Definition of the Problem

The measurement of EPs is complicated by the fact that they are of much smaller magnitude than the on-going e.e.g.. Improvement of signal-to-noise ratio has conventionally been achieved by collecting a large ensemble of responses or EPs to repeated identical stimuli and performing ensemble averaging. Under assumptions that EP is deterministic, background e.e.g. is a zero-mean random process and that the two are uncorrelated, it can be shown that ensemble averaging will improve the signal-to-noise ratio by a factor equal to the square root of the number of stimulus repetitions.

However, there is no a priori reason to assume that responses evoked by a series of identical stimuli will themselves be identical (implying EP is deterministic). The effects of habituation, fatigue or distraction are known to alter the characteristics of some evoked potentials, particularly late waves. Averaged EPs may also be misleading since they do not reflect the momentary changes in the excitability of the central nervous system that are crucial to the normal response of the organism to environmental change. Only those components of the individual responses that are common to all responses will be brought out by averaging. In the study of physiological systems, the "complete" response to a stimulus is at least as important as the common elements in all responses to a large sequence of identical stimuli.

In using ensemble averaging technique, the requirement of repetition imposes a time lag before the EP is available. This time lag together with the loss of transient responses precludes the effective use of the EP technique in monitoring patients in the intensive care unit and during neurosurgery. In addition, the loss of transients and habituation to successive stimuli make this technique less appropriate in the study of cognitive processes using the contingent negative variation or P300 (two types of EPs).

Adaptive filtering techniques have become very popular in the past five or ten years in the signal processing field. Adaptive noise cancellation, where information from pre-stimulus e.e.g. can be used to cancel e.e.g. in the post-stimulus data, thereby enhancing the signal-to-noise ratio, holds promise. The adaptive technique used in the cancellation scheme is of primary importance in attaining acceptable signal-to-noise ratios.

Therefore, having seen that the assumption that EP is deterministic is untenable and that ensemble averaging as a method of improving signal-to-noise ratio is undesirable, we arrived at our first research objective.

Research Objective (1) :-

Derive an adaptive noise cancellation filter to estimate the evoked potential.

In present day neurological practice, the clinician identifies the peaks of the EP estimated by conventional averaging or by some new method by visual inspection and classifies it as normal or abnormal based on its appearance. The quantitative guidelines in this classification procedure are mainly the latencies of the peaks. Selective attenuation of some peaks are also indicative of abnormality. The main motivation for automatic pattern recognition and classification of EPs arises out of a need to supplement the clinician's visual analysis with a more objective method. In the process of pattern recognition and classification, it is also desirable to build up a description of the pattern under consideration. This allows us to give reasons in terms of peak latencies and amplitudes (the parameters used by the clinician) for classification as normal or abnormal. The objective criteria used by the automatic classifier allows the comparison of peak latencies and amplitudes between subjects and centers. The unsupervised nature of the pattern recognition system will make it ideal for screening at remote locations, continuous monitoring in intensive care situations and during surgical procedures.

Research Objective (2) :-

A pattern recognition method for classification and generation of description of EP.

These general research objectives are precisely defined in section 5.3 and section 6.2 respectively, stating explicitly the assumptions made and other theoretical considerations.

1.2 Proposed Solutions

We give an outline of the proposed solutions to the two research questions raised in the previous section. The details of the methods used are the subject matter of Part – B (Adaptive Filtering) and Part – C (Pattern Recognition).

(1) Signal Estimation

The adaptive noise cancellation method was proposed by Widrow et.al. (1975) to extract signal from noise. It is assumed that a reference source of noise is available which is correlated only to the noise in the signal plus noise source (called primary source). An adaptive filter modifies the reference noise such that the expected value of the squared error between the modified reference noise and the noise in the primary source is minimized. Because of the eigenvalue sensitivity problems of the Least-Mean-Square (LMS) algorithm that Widrow and others used for the adaptive noise cancellation, we derived a Weighted exact Least Squares Lattice (WLSL) in the joint estimation form as a generalization of the exact LSL derived by Lee et.al.(1981) and Haykin (1986) for adaptive noise cancellation. Post stimulus EP plus e.e.g. is the primary source and pre stimulus e.e.g. is the reference source (alternatively, concurrent e.e.g. from a different location 'uncontaminated' by EP could be the reference source). We show by simulations and tests with real data that EP can be estimated without having to repeat the stimulus. To distinguish it from conventional averaged EP, we call EP estimated by this method, 'Single Stimulus Evoked Potential' or SSt EP.

(2) Pattern Recognition

Considering the structural specificity of the EP waveforms, the procedure followed by the clinician can be best automated by syntactic pattern recognition. This method also builds up a description of the pattern as a natural part of the process. The EP waveform is first preprocessed by zero-phase bandpass filter (thus avoiding any phase distortion) to remove artifacts and noise. A finite-state grammar which can identify a 'hill' is used in the pattern recognition procedure. The parse table that is generated is a complete description of the EP waveform. The salient features of the EP, such as location of the peaks, can be identified and used for classification. A data base of normal EP parameters is generated and referenced for classification and is updated suitably. Acceptance criteria are optimized for clinical applications during the training run of the classifier. The method was tested in great detail using real EPs of unknown diagnosis obtained from actual patient records.

The new signal estimation technique and the pattern recognition technique are quite general in nature in that they can be applied to any of the well-known EPs, such as visual, auditory, somatosensory or cognitive, after suitable minor modifications. The weighted exact least square lattice algorithm that is derived here is the first of its kind and is the most general form. Weights can be calculated on-line to give the algorithm desirable properties such as statistical robustness.

In this thesis, when application considerations are discussed, Brain Stem Auditory Evoked Potential (BSAEP) is taken as the example.

1.3 Thesis Outline

Background material relevant to this thesis is presented in Part - A which comprises chapters 2 and 3. We begin Chapter 2 with the physiology of BSAEP including a brief introduction to various electrophysiological theories of origin of EPs. After giving the physiological and clinical significance of BSAEP, we arrive at a simplified engineering model of signal generation, which we will refer to throughout the rest of the thesis: In Chapter 3, after a quick overview of EP acquisition techniques, we present a detailed review of literature relevant to signal processing and pattern recognition of EPs.

Part - B deals with the adaptive filtering technique used for single stimulus EP extraction. In Chapter 4, we discuss adaptive noise cancellation and derive the weighted exact least squares lattice structure. In Chapter 5, general simulation studies are considered. Comparison of normalized exact least squares lattice algorithm and WLSL is made and results of real data test are given.

The syntactic pattern recognition method is discussed in Part - C of the thesis. In Chapter 6, the theory, application considerations and results of pattern recognition and classification are given.

We conclude the thesis with a detailed discussion and criticism of the research work. Recommendations for future research are included in the last section.

The overview section at the beginning of each chapter gives a more detailed account of the contents. We thus leave further details to them.

P A R T – A

BACKGROUND

CHAPTER 2

PHYSIOLOGY OF EVOKED POTENTIALS

2.1 Overview of the Chapter

As part of the background material, we take a look at the anatomical and physiological implications of evoked potentials in general and brain stem auditory evoked potential (BSAEP) in particular. In section 2.2, general theories of the origin of evoked potentials are discussed. The clinical utility of evoked potentials is then discussed in section 2.3 to justify the present effort. A simplified engineering model of evoked potential generation is arrived at in section 2.4, which takes into consideration most of the relevant physiological details.

2.2 Origin of Evoked Potentials

The search for a buried message in the electroencephalogram (e.e.g.) has been going on, right from the days of Hans Berger (1929), the first person to record the e.e.g.. After a period of careful scrutiny of the time and frequency domain characteristics of e.e.g., interest has shifted to responses evoked by stimuli to the human sensory system called the evoked potential (EP). This interest is partly justified by the knowledge that EP reflects in some fashion, the functional aspects of the specific neuroanatomical pathways activated by the stimulus.

The myriads of external stimuli that we continuously receive along with endogenous stimuli activate the human brain, the resultant of which can be measured on the scalp as e.e.g.. This is considered to be the spatial and temporal summation of the post-synaptic potentials arising from billions of neurons in the human brain (Kiloh et al., 1981). The e.e.g. is usually of the order of 100 microvolts. A specific stimulus presented to the sensory system activates only a small proportion of this neuronal population. Not very surprisingly, its reflection on the scalp which is the EP, is difficult to detect as its amplitude ranges from sub-microvolt to tens of microvolts. Dawson (1947) estimated the EP by superposing an ensemble of responses on a photographic plate, thereby improving the signal to noise ratio. Since that time, dedicated hardware which essentially performs the same averaging technique has been used to estimate the EP. Figure 2.1 illustrates this paradigm where one specific type of EP, the brain stem auditory EP (BSAEP) is shown.

Of the many classification systems of EPs, the one we are interested in is based on its origin within the brain. The first one is the sensory or short latency or far-field EP and the other is the long latency or near-field EP. Among the various theories of the origin of EP (see Childers, 1977 for an excellent review), the short latency EP can be assumed to be an example of "field theory model" and long latency EP, that of "complex cortical connection model". In the case of short

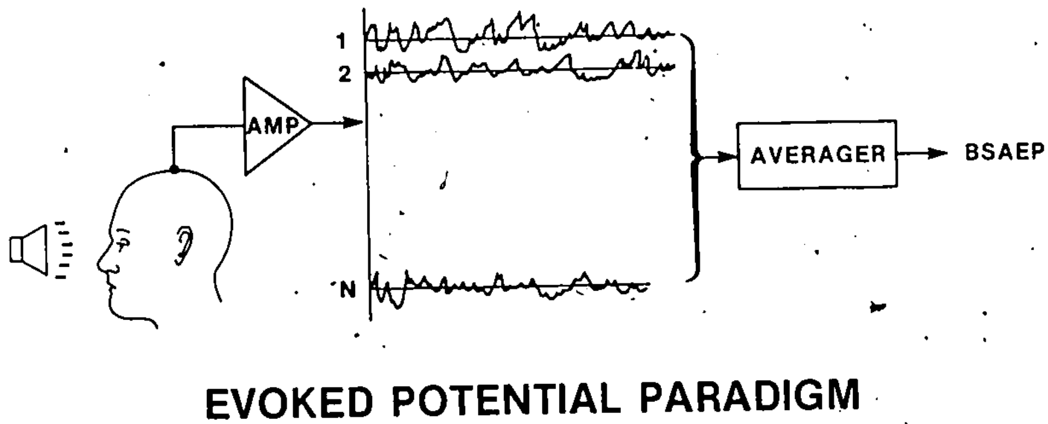

Figure 2.1 Evoked potential paradigm.

latency EP, deep structures are activated (in-vivo current source density analysis done on mice brain by Madhavan (1980) lends some credence to this) and by volume conduction is seen almost equally at all points of the scalp. However, for the long latency EP, cortical areas specific to a certain stimulus close to the scalp are activated and the ionic currents propagating through the pathways give rise to potentials in localized areas of the scalp. The e.e.g., having a similar origin, it can be hypothesized that e.e.g. may be more correlated to long latency EP than to short latency EP.

Brain stem auditory EP (BSAEP) is an example of short latency EP. In response to auditory stimulus, the VIIIth nerve and brain stem areas of the auditory pathway are activated and its response can be seen all over the scalp in the first 10 milliseconds after the stimulus (hence the name short latency EP). Given the fact that in this short period, the generators of e.e.g. in the cerebral cortex could have hardly responded to the stimulus (ie., e.e.g. "desynchronized"), BSAEP is probably volume conducted to the scalp and is uncorrelated to the on-going e.e.g.. BSAEP has strong correlation with the integrity of the underlying anatomical structures (Jewett and Romano, 1972). It is usually recorded from the vertex (Kiloh et al., 1981) and is of very small amplitude. Often, an ensemble as large as 2000 has to be averaged to enhance the signal to noise ratio sufficiently. Figure 2.2 shows an idealized BSAEP.

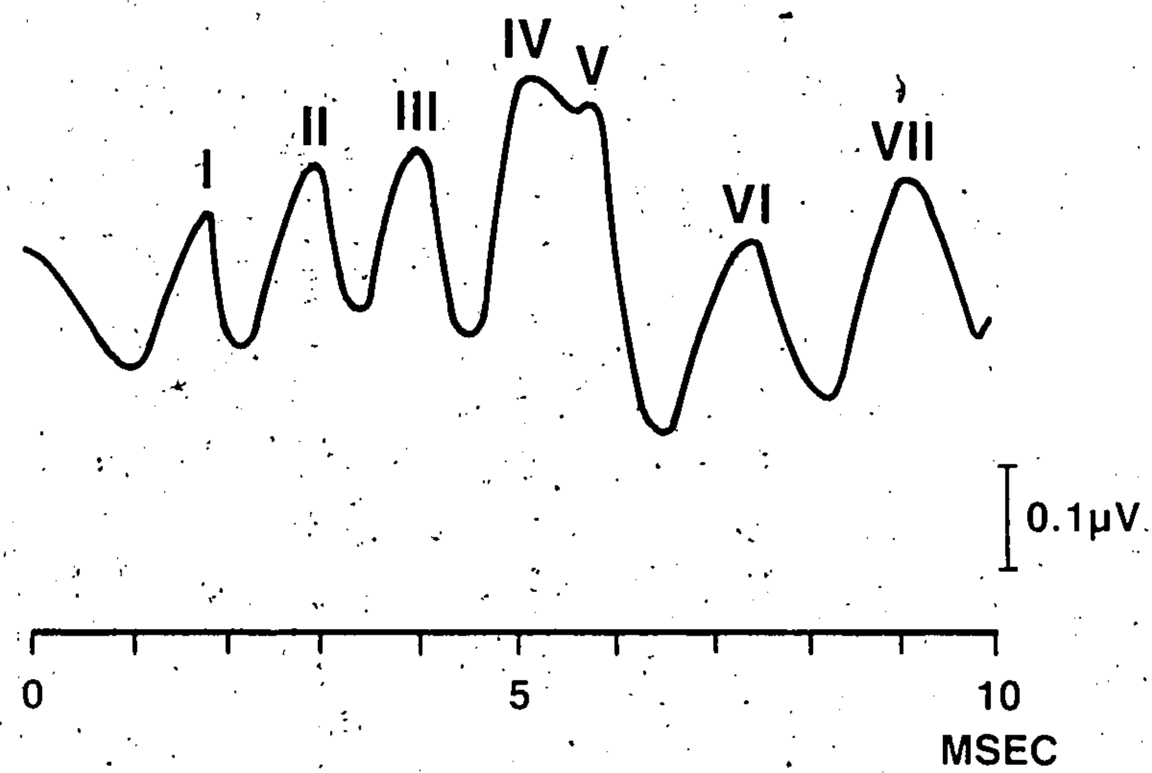

Figure 2.2 Idealized brain stem auditory EP.

2.3 Physiological Significance of Evoked Potentials

Evoked potential tests provide a sensitive and quantitative extension to the clinical neurological examination (Chiappa, 1983). They have the ability to demonstrate abnormal sensory system function when the neurological examination is equivocal. EPs also help define the anatomical distribution of a disease process and objectively monitor changes in a patient's status over time. It is the changes in shape, amplitude and latency (duration from stimulus on-set to the event) of the peaks in the EP that give the clinician this objective information. In the case of most EPs, it is primarily the changes in latency that give information about the abnormality to the clinician (Kiloh et al., 1981; Chiappa, 1983).

BSAEPs are remarkably constant in the same patient. They can therefore be used to assess the integrity of the brainstem and hence the prognosis in unconscious patients. These potentials have also been useful in the diagnosis of acoustic neuromas and other clinical conditions. They may also be used to assess the physiological effects of drugs or neurostimulation. A detailed treatment of the clinical utility of this and other EPs are given in Kiloh et al. (1981).

From the foregoing discussion, it is clear that EPs have tremendous clinical as well as research utility. A major stumbling block in its more efficient use has been the need to repeat the stimulus to generate a large enough ensemble so that averaging will improve the signal to noise ratio sufficiently. Repetition of the stimulus entails a delay in obtaining the EP as well as confounding the measurement by the effects of habituation and fatigue. In this thesis, we undertake an investigation of different methods of improving the signal to noise ratio so that stimulus repetition can be minimized or eliminated.

2.4 A Working Model

To facilitate the development of signal processing methods in the succeeding chapters, we will develop a model for the generation of the EP.

In published literature, e.e.g. has been effectively modeled as a time series. Work of Kaveh et al. (1978) and Rauner et al. (1983), for example, shows that an autoregressive (AR) time series model of order 8 to 10 is sufficient to model e.e.g.. Our own experience in the past years have shown that filtered e.e.g used in BSAEP work can be modeled as an AR model of order 6. Hence, in general, e.e.g. can be considered to be a general time series which is the output of an autoregressive, moving average (ARMA) filter excited by a white, Gaussian noise sequence. This is shown in figure 2.3 where 'v' is the e.e.g. and 'w' is the white, Gaussian noise. When a stimulus is presented, the neuronal pathway stimulated generates a response which we have called SSt EP. It is well known that when the stimulus intensity is doubled, the respone is not doubled (Kiloh et al., 1980), indicating a saturation-type non-linearity in the system. As a complex interconnection of neurons in the

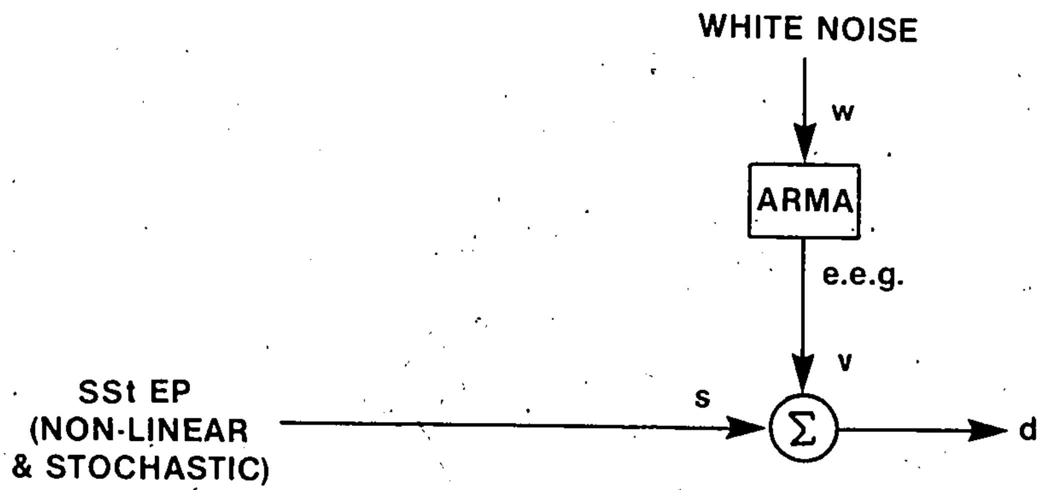

Figure 2.3 A working model of EP generation.

stimulated pathway gives rise to SSt EP, we see shape and latency variations or "jitter" (Aunon et al., 1981) in SSt EP, which makes it non-deterministic. This non-linear and stochastic signal, SSt EP is 's' in figure 2.3. Since, in the case of short latency EPs, SSt EP is the potential generated by thalamic and lower regions and e.e.g. is the potential generated by cerebral cortical regions, they can be considered mutually uncorrelated and the total response, 'd' in figure 2.3 can be considered as their additive combination at the scalp.

Assumptions

Keeping in mind the signal processing technique that we are going to employ (the subject matter of chapters 4 and 5), we make the following minimally restrictive assumptions. Referring to figure 2.3 where 'd' is the observed time series in response to the stimulus, 'v' is the on-going e.e.g. and 's' is the actual response of the activated neuronal pathway (which we call single stimulus EP or SSt EP), we assume that -

1. the **signal** (SSt EP or 's') and **noise** (e.e.g. or 'v') **combine additively** to give the potential measured from the scalp, which we will call the response or 'd'.
2. the **noise** (e.e.g. or 'v') is a **correlated time series.**
3. no restrictions are placed on the characteristics of the **signal** (SSt EP or 's'). In general, 's' is **non-linear and stochastic.**
4. **signal and noise are not correlated.**

In general EP methodology, assumptions 1 and 4 are always employed (eg., Aunon et al., 1981; Childers, 1977; Rauner et al., 1983; de Weerd and Martens, 1978; Kaveh et al., 1978). In assumption 2, we do not place the usual restrictive condition that the time series be stationary (see the references above) even though e.e.g. can sometimes be considered stationary for epochs shorter than 12 seconds (Cohen and Sances, 1977). Assumption 3 is entirely general without the usual deterministic and linear constraints. As we shall see in chapters 4 and 5, these more realistic assumptions can be accomodated by the signal processing technique that we use, which is adaptive noise cancellation using weighted least squares lattice algorithm.

In the next chapter, we provide a comprehensive review of the signal processing and pattern recognition techniques used in EP processing.

CHAPTER 3

METHODS OF PROCESSING EVOKED POTENTIALS

3.1 Overview of the Chapter

Evoked potentials have been in use for almost four decades now. Many methods from photographic superimposition and visual analysis to specially designed computer systems with sophisticated signal processing techniques have been used in its acquisition and processing. In this chapter, we review in detail the present day techniques that are widely used and are related to our research. In section 3.2, we give a brief outline of electrode requirements, electrode montages (10-20 system), amplifier requirements, etc. for the acquisition of BSAEP. Section 3.3 is a comprehensive review of the signal processing techniques used for evoked potential studies. We look at the assumptions involved, basic theory and applications of the conventional averaging technique and variants thereof, types of optimal Wiener filtering, time varying filtering and predictor-subtractor-restorer filtering. In section 3.4, we review the statistical and syntactic methods of pattern recognition as applied to evoked potentials.

3.2 Acquisition of Evoked Potentials

In this section, we take a brief look at hardware requirements for EP acquisition. It is to be noted that the requirements vary depending on the type of EP being processed. References like Kiloh et al. (1981) or Chiappa (1983) give the technique of measurement for all types of EPs. Here, we concentrate on the requirements for the acquisition of BSAEP since, in the succeeding chapters, our signal processing and pattern recognition techniques are developed with BSAEP as the example for application.

BSAEP is recorded from the vertex (C_z) on top of the skull with reference electrodes on the earlobes (A_1-A_2) (see figure 2.1). The locations on the scalp are determined under the International 10-20 system (Kiloh et al., 1981). Conventional e.g. gold disc electrodes are used. Although gold electrodes have poor d.c. and low frequency response (Geddes and Baker, 1975), it is entirely adequate for BSAEP since most of its energy lies in the 400-1500 hz band (Madhavan et al., 1983). The stimulation used is auditory biphasic clicks lasting for 200 microseconds, delivered to the ear through headphones. Stimulation is repeated at the rate of 10 hz. The volume of the clicks is usually 60 to 75 db above hearing threshold.

Differential amplifiers with high common mode rejection ratio and low noise are employed for amplification, considering the small amplitude of the signal (less than 1 microvolt). Amplification of the order of 1,000,000 is usually required. Since the energy of the signal (400-1500 hz) and principal noise (maximum around 10 hz) lie in widely separated frequency bands (Boston, 1981), bandpass filtering can be gainfully employed. In clinical work, bandpass filtering of 150 to 3000 hz is used. The consequence of this is that whereas with an unfiltered response, the signal to noise ratio may be as adverse as -20 db, bandpass filtering improves it to the order of -5 db to 0 db. The price to be paid is the phase distortion and hence latency distortion in the estimated EP (Boston and Ainslie, 1980). However, 150 to 3000 hz is considered quite acceptable at low orders of the filter.

The first 10 milliseconds after stimulus onset is analog to digital converted at rates of 10 to 50 Khz which satisfies the Nyquist rate. Repeated responses are acquired and ensemble averaged by a computer. Averaging around 2000 responses enhances the signal to noise ratio sufficiently and the estimated EP is displayed on the video monitor. The clinician or the technician then visually identifies the peaks and the normalcy of the EP is determined.

3.3 Current Signal Processing Techniques

Averaging is the true and tested method in EP processing. Desirability of quicker and better estimates of EP has spawned considerable research into alternate methods. A consistent feature in all these approaches is the aim of reducing the number of repetitions required before an acceptable EP estimate is obtained.

3.3.1 Conventional Averaging

Referring to figure 2.3 in the last chapter, we can write the equation for the observed response, d on the scalp at time instant i for j th repetition of the stimulus with the assumptions that signal and noise is additive, signal, s is deterministic, noise, v is a zero-mean random process and that signal and noise from trial to trial are uncorrelated.

$$d_j(i) = s(i) + v_j(i) \qquad j=1,2,\ldots,N$$

Expected value of d(i) then is

$$E[d(i)] = E[s(i)] + E[v(i)] = s(i)$$

which means that ensemble average is the estimate of single stimulus EP (SSt EP). With the additional assumption that the average noise power is the same in each realization in the ensemble, it can be shown that signal to noise ratio is improved proportional to the square root of the number of repetitions, N. However, it is easy to see that the assumption that the signal is deterministic is untenable because we cannot expect a complex system like neuronal pathways to respond identically to each stimulus. Thus, in the process of averaging, a great deal of potential information about the individual responses or SSt EP is lost. In any case, averaging is the most popular method of EP processing, mainly because of its conceptual and computational simplicity.

3.3.2 Improvements to Averaging

Implied in the assumption that signal (or SSt EP) is deterministic is that its onset delay after the stimulus is invariant. However, in practice we do see that there is jitter in the response. Many methods have been proposed to accomodate this factor.

Basar (1980) used a method of selective averaging where each response is visually inspected and 'acceptable' ones are averaged. This is not so much to compensate for jitter in the response than to eliminate responses with obvious artifacts which could deteriorate the ensemble average.

In cross-correlation averaging (Woody, 1967), each response is cross-correlated with a template (typically a conventionally averaged EP) and shifted till cross-correlation is maximized. Such aligned responses are ensemble averaged to give the estimate of SSt EP. The fact that SSt EP over different repetitions may have shape variations, the background e.e.g. may have shapes similar to the template and data lengths are not long enough to calculate reliable statistical expectations introduce errors in this estimation procedure.

Aunon and McGillem (1975) proposed a method wherein each individual peak of the SSt EP is aligned before averaging. Individual responses are pre-processed by a minimum mean square error filter using a previously averaged EP as the desired response. Each peak is selected by matching it with a template and a histogram of the number of peaks at each latency is obtained. By statistical procedures, peaks are assigned to each latency interval. In each of these intervals, detected peaks are aligned and averaged. Although the minimum mean square error filter can distort the signal to some extent and similarity of SSt EP and e.e.g. components can lead to incorrect alignment, this method performs better than conventional averaging.

These methods are reviewed in more detail in Aunon et al. (1981). A few more methods of improvement to conventional averaging like median averaging (Borda and Frost, 1968), frequency domain averaging (Auerbach and Haber, 1974; Rodriguez et al., (1981)) are available in the literature.

3.3.3 Wiener Filtering Techniques

Based on Wiener's optimal filter theory (1949), which minimizes the mean square of the estimation error, Walter (1969) proposed its use in EP processing. Doyle (1975) proposed some modifications. In a series of articles, de Weerd (1981) explored this technique and extended its application to EPs. Yu and McGillem (1983) have approached the optimal filtering problem from least squares estimation theory.

As we have seen earlier, the response, d on the scalp to a stimulus can be written as the sum of SSt EP, s and e.e.g., v.

$$d(i) = s(i) + v(i)$$

From Wiener filter theory, we can write the transfer function of the optimal filter which will estimate the signal, s(i) as (see Kailath, 1981)

$$H(\omega) = \frac{S_{ss}(\omega)}{S_{ss}(\omega) + S_{vv}(\omega)}$$

where $S_{ss}(\omega)$ and $S_{vv}(\omega)$ are the power spectra of SSt EP and e.e.g.. The above formulation assumes independence between s(i) and v(i) and their stationarity. As the power spectra of these quantities are not known, we will have to work with the average spectra of individual responses, d(i) and that of its ensemble average, $\bar{S}_{dd}(\omega)$ and $S_{\bar{d}\bar{d}}(\omega)$ respectively. For filtering the ensemble averaged EP (Doyle, 1975), it can be easily shown that the appropriate filter transfer function is

$$H_a(\omega) = \frac{N}{N-1}\left[1 - \frac{1}{N}\,\frac{\bar{S}_{dd}(\omega)}{S_{\bar{d}\bar{d}}(\omega)}\right]$$

The fact that e.e.g. is sometimes non-stationary and that estimates of spectra rather than true spectra are used compromise the effectiveness of this filtering scheme.

3.3.4 Time Varying Filtering

de Weerd (1981) and de Weerd and Kap (1981) assume signal and noise to be non-stationary and suggest a time varying filtering technique. It can be characterised as a bank of constant relative bandwidth filters (bandwidth proportional to the centre frequency of the band), followed by time-varying attenuators which are controlled by the estimated time-varying signal to noise power ratio in the corresponding frequency bands. The outputs of these attenuators are summed to give the estimate of the signal. Even though this filter is available on Nicolet Pathfinder EP equipment as a software package (de Weerd, 1984), there is not much published information regarding its performance.

3.3.5 Predictor-Subtractor-Restorer Filtering

Kaveh et al. (1978) made clever use of the information in pre-stimulus e.e.g.. If indeed, the pre-stimulus e.e.g. is highly correlated with the e.e.g. in the post-stimulus, as will be the case for short-latency EPs, that information can be used in extracting SSt EP from the response.

The post-stimulus response is filtered using the autoregressive (AR) model of pre-stimulus e.e.g.. The result is a sum of SSt EP distorted by this filter (coefficients of which are known, the AR parameters) and whitened (assuming pre and post-stimulus e.e.g. are the same) post-stimulus e.e.g.. This mixture of signal and noise lends itself readily to Wiener filtering. The distorted signal obtained is inverse filtered using the known AR coefficients to obtain an estimate of the SSt EP. Some improvements to this method, mainly in the Wiener filtering step, have been suggested by Rauner et al. (1983).

It may be noted that instead of the block method of processing used by Kaveh et al., an adaptive structure involving an AR lattice predictor, k-step adaptive line enhancer and inverse AR lattice will be an attractive alternative.

The idea of using pre-stimulus e.e.g. is excellent but the choice of application to visual EP that Kaveh et al. made may be inappropriate (correlation between pre and post stimulus e.e.g. in the case of this near-field EP may not be high and hence not whiten the post-stimulus e.e.g.). The assumption of stationarity of the data and the rather simple signal spectrum that Kaveh et al. chose (a low-pass function) in the formulation of the Wiener filter may have contributed to the rather unspectacular results.

As we have seen in this section, the field of EP signal processing has a rich history. The non-linearity and non-stationarity of the system under study is a major stumbling block in these efforts. However, the clinical and research importance of estimating single stimulus EPs (as opposed to averaged EPs) cannot be overstated. In our approach, we borrow some ideas that were discussed here (like the use of pre-stimulus e.e.g.) and make use of the new developments in the field of signal processing like adaptive filtering, lattice structures, etc.. This is the subject matter of PART B of this thesis.

3.4 Current Pattern Recognition Techniques

The main motivation for developing pattern recognition and classification schemes for evoked potential arises out of a need to supplement the visual analysis of EP with a more objective and less tedious method. In the absence of methods to estimate single stimulus evoked potentials, some classification methods have utilized features obtained from the complete response (ie., SSt EP plus e.e.g.). The features used in these statistical classification methods may have no direct relationship to features like peak latencies and shapes used in usual EP practice. Nevertheless, the ability to automatically classify averaged or SSt EP is a very useful addition to EP methodology.

The approach we have used is called syntactic pattern recognition where information from the structurally specific shape of the averaged or SSt EP is utilized, similar to the clinician's approach. However, for the sake of completeness, we will review briefly the statistical pattern recognition techniques used for evoked potential classification

3.4.1 Statistical Methods

In the method of statistical pattern recognition, for unlabelled samples, clustering is done to find class defining information after extracting features using, say, Karhunen-Louve expansion. Classification is done based on distance metrics, probability measures or by directly deriving a discriminant function from the features.

In the literature, we see a variety of simple features used for pattern recognition. Aunon and McGillem (1982) used the amplitudes at each instant of the waveform as the features. Ono et al. (1981) used the AR coefficients of the EP model as their features. Others have used spectral components estimated by maximum entropy method (Lam et al., 1982) or by Walsh transform method (Larsen and Lai, 1980). These can be looked upon as choices sub-optimal to Karhunen-Louve (K-L) expansion since K-L expansion uses basis vectors estimated from the data whereas other transform methods use pre-selected basis vectors (Ahmed and Rao, 1975). Most of the classifiers used are either based on probabilistic measures (Aunon and McGillem, 1982) or discriminant functions (Aunon et al., 1982; Ono et al., 1981).

Childers et al. (1982) proposed a novel way of feature selection using segmented data. Instead of selecting maximum eigenvalue eigenvectors as the basis, a figure of merit is used to select the basis vectors. Using a linear discriminant function, they have achieved very low probability of error (sometimes zero) of classification.

3.4.2 Syntactic Methods

The method of syntactic pattern recognition was developed by Fu (see Fu, 1982 for a complete exposition of the method) from principles of Theoretical Linguistics and Computer Science. Tou and Gonzalez (1974) also give a clear description of the method.

When the pattern to be recognized and classified is structurally specific, the syntactic approach offers a computationally simple and conceptually straight-forward method. In the case of BSAEP, we have seen in figure 2.2 that it has a well defined shape. Most EPs exhibit this property. This being the case, the syntactic method further offers the advantage of constructing a description of the pattern under study which, as we will see in PART C of the thesis, is a major advantage in clinical implementation. Besides, this method uses the very parameters used by the clinician such as peak shapes, amplitudes and latencies, to arrive at the classification.

In the literature, we find instances of syntactic methods being used for e.e.g. analysis (Bourne et al., 1981; Bourne et al., 1980), where features extracted from the e.e.g. power spectrum are the primitives. We do not find in the literature, any application of syntactic pattern recognition to evoked potential data. In PART C of the thesis, we will discuss the theory of the syntactic method and give details of our work in the area.

PART B

ADAPTIVE FILTERING

CHAPTER 4

THEORY OF ADAPTIVE NOISE CANCELLATION USING WEIGHTED EXACT LEAST SQUARES LATTICE

4.1 Overview of the Chapter

As an alternative to conventional averaging and the other methods discussed in Chapter 3, we propose adaptive noise cancellation using weighted exact least squares lattice algorithm. The theory of adaptive noise cancellation is presented in section 4.2. In section 4.3, we derive the new weighted exact least squares lattice algorithm in the joint estimation form for adaptive noise cancellation. This is the most general form of the lattice in that weighting can be attached to the data depending on a priori information available about the process(es), statistical robustness issues, etc..

4.2 Theory of Adaptive Noise Cancellation

Even though adaptive noise cancellation has been sporadically used since the late 1950's, this technique has become much more popular since the landmark paper of Widrow and colleagues in 1975. In that paper, they give a thorough treatment of the theory, limitations and applications of adaptive noise cancellation (Widrow et al., 1975).

One of the classical problems in signal processing is estimating a signal corrupted by additive noise. There is a large body of literature dealing with optimal Wiener filtering to achieve this objective. In section 3.3.3, we have reviewed some applications of Wiener filtering using fixed filters in evoked potential research. A priori information regarding signal and noise processes are required in implementing such filters. An alternative that has gained much popularity in recent years has been filters that are data adaptive. Such filters automatically adapt to the incoming data to achieve the objective of filtering out the noise. Whether fixed or adaptive, such filters require either a priori information regarding the processes or a training sequence to tune the adaptive filter.

Adaptive noise cancellation is an alternative where an additional source of noise, called "reference input", correlated to the noise corrupting the signal, is made use of. The source of signal along with the additive noise is called the "primary input". The reference input is filtered adaptively to give an estimate of the primary input noise. This estimate can then be subtracted from primary input to achieve levels of noise reduction with little signal distortion usually impossible with fixed filters, if the conditions are appropriate.

The configuration for adaptive noise cancellation is shown in figure 4.1. $u(i)$ and $d(i)$ are the reference and primary inputs respectively. Primary input is the sum of the signal to be estimated $s(i)$ and noise corrupting it, $v(i)$ which is correlated to $u(i)$. Reference input $u(i)$ is adaptively filtered to yield an estimate, $\hat{v}(i)$ of noise in the primary input, $v(i)$. $\hat{v}(i)$ is subtracted from primary input, $d(i)$ to give the error, $e(i)$. The adaptive filter coefficients are automatically adjusted to minimize this error.

The assumptions involved in adaptive noise cancellation are:

(i) Primary input contains signal, $s(i)$ corrupted by additive noise, $v(i)$.

(ii) $s(i)$ is uncorrelated to $v(i)$ and reference input, $u(i)$.

(iii) $u(i)$ is correlated to $v(i)$.

The adaptive noise cancellation problem as stated above can be considered as a linear least squares estimation problem where we estimate the parameters of the multiple linear regression model (Haykin, 1986) relating $u(i)$ to $d(i)$ (in fact, $u(i)$ to $v(i)$ due to the above mentioned assumption (ii)). Then, using an Mth order linear transversal filter, figure 4.1 can be represented as in figure 4.2 where z^{-1} is the backward shift operator. This problem is also called joint process estimation because we estimate one process $\{d(i)\}$

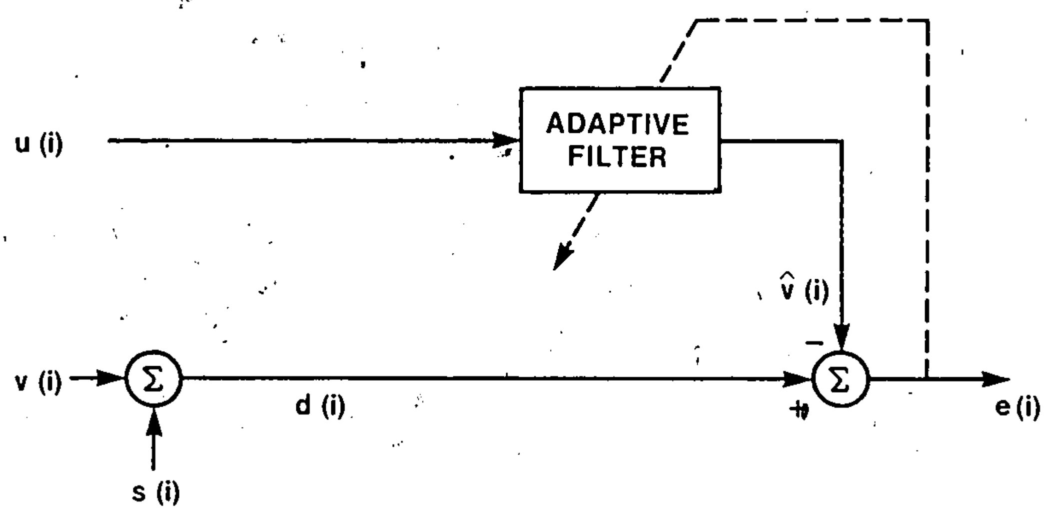

Figure 4.1 Adaptive noise cancellation configuration.

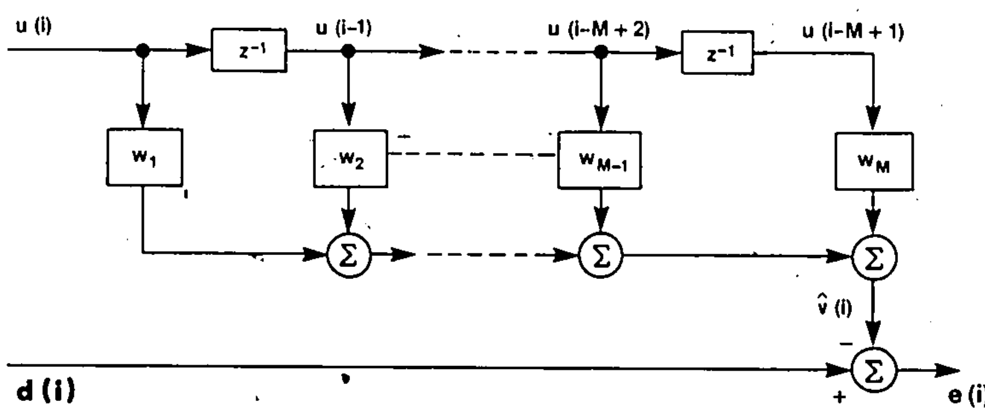

Figure 4.2 Transversal filter structure for ANC.

(actually {v(i)}) from observations of a related process {u(i)} by embedding them into a joint process {d(i),u(i)} (Lee, 1980).

Let us consider the real, scalar case of the problem proposed above. To find the least squares estimates of the parameters of the multiple linear regression model, choose the tap weights, w_k such that the performance criterion,

$$E(w_1, \ldots, w_M) = \sum_{i=i_1}^{i_2} |e(i)|^2 \tag{4.2.1}$$

is minimized, where

$$e(i) = d(i) - \sum_{k=1}^{M} w_k\, u(i-k+1), \qquad i_1 \le i \le i_2 \tag{4.2.2}$$

Define the following M vectors:

(i) Reference input vector,

$$\underline{u}(i) = \begin{bmatrix} u(i) \\ u(i-1) \\ - \\ - \\ - \\ u(i-M+1) \end{bmatrix}$$

(ii) Tap weight vector,

$$\underline{w} = \begin{bmatrix} w_1 \\ w_2 \\ - \\ - \\ - \\ w_M \end{bmatrix}$$

Equation for error, e(i) in (4.2.2) can be re-written as:

$$e(i) = d(i) - \underline{u}^T(i)\, \underline{w}, \qquad i_1 \le i \le i_2.$$

for all the (i_2-i_1+1) time instants. To write all the (i_2-i_1+1) equations in a compact form, define the following (i_2-i_1+1) vectors:

(i) Error vector,

$$\underline{e} = \begin{bmatrix} e(i_1) \\ e(i_1+1) \\ - \\ - \\ - \\ e(i_2) \end{bmatrix}$$

(ii) Primary input vector,

$$\underline{d} = \begin{bmatrix} d(i_1) \\ d(i_1+1) \\ - \\ - \\ - \\ d(i_2) \end{bmatrix} = \begin{bmatrix} s(i_1) \\ s(i_1+1) \\ - \\ - \\ - \\ s(i_2) \end{bmatrix} + \begin{bmatrix} v(i_1) \\ v(i_1+1) \\ - \\ - \\ - \\ v(i_2) \end{bmatrix}$$

or $\underline{d} = \underline{s} + \underline{v}$ where $\underline{s}$ and $\underline{v}$ are as shown above and $s(i)$ is the signal and $v(i)$ is the additive noise in the primary input (refer figure 4.1). Now, all the (i_2-i_1+1) equations can be written in a compact form as below

$$\underline{e} = \underline{d} - A\,\underline{w} \tag{4.2.3}$$

where A is a $((i_2-i_1+1)\times M)$ Toeplitz matrix.

$$A = \begin{bmatrix} \underline{u}^T(i_1) \\ \underline{u}^T(i_1+1) \\ - \\ - \\ - \\ \underline{u}^T(i_2) \end{bmatrix}$$

Equation (4.2.1) can be rewritten as

$$E(\underline{w}) = \underline{e}^T\,\underline{e} \tag{4.2.4}$$

To find the least squares estimates, $\hat{\underline{w}}$ of $\underline{w}$, vary $\underline{w}$ such that $E(\underline{w})$ is a minimum. Equivalently, if we differentiate (4.2.4) with respect to (w.r.t.) $\underline{w}$ and equate to zero, the solution is the estimate we desire, $\hat{\underline{w}}$.

$$\frac{\partial}{\partial \underline{w}} E = 2\left(\frac{\partial \underline{e}}{\partial \underline{w}}\right)^T \underline{e}$$

But $\partial \underline{e}/\partial \underline{w} = -A$ from equation (4.2.3).

$$\therefore \quad \frac{\partial}{\partial \underline{w}} E = -2 A^T \underline{e} = 2(A^T A \underline{w} - A^T \underline{d}) \tag{4.2.5}$$

To find $\hat{\underline{w}}$, i.e., value of $\underline{w}$ when $E(\underline{w})$ is a minimum, equate (4.2.5) to zero to get

$$A^T A \hat{\underline{w}} = A^T \underline{d} \tag{4.2.6}$$

This is the deterministic system of Normal Equations. The minimum error using the least squares estimate of the tap-weights, $\underline{w}$, is

$$\begin{aligned} \underline{e}_{min} &= \underline{d} - A \hat{\underline{w}} \\ &= \underline{s} + (\underline{v} - A \hat{\underline{w}}) \end{aligned}$$

Second term on the right hand side represents the measurement noise in the multiple linear regression model relating u(i) and v(i). Hence, better the model, smaller the second term will be and $\underline{e}_{min}$ will be the "best" estimate of signal $\underline{s}$ in the least squares sense.

The minimum value of sum of error squares is

$$\begin{aligned} E_{min} &= \underline{e}_{min}^T \underline{e}_{min} \\ &= \underline{d}^T \underline{d} - \underline{d}^T A \hat{\underline{w}} \end{aligned} \tag{4.2.7}$$

Taking (4.2.6) and (4.2.7) together, augmented normal equations can be written as

$$\begin{bmatrix} \underline{d}^T \underline{d} & \underline{d}^T A \\ A^T \underline{d} & A^T A \end{bmatrix} \begin{bmatrix} 1 \\ -\hat{\underline{w}} \end{bmatrix} = \begin{bmatrix} E_{min} \\ \underline{0}_M \end{bmatrix} \tag{4.2.8}$$

where $\underline{0}_M$ is a null vector of order M.

Widrow et al. (1975) give an alternate exposition from stochastic point of view. Instead of the least squares algorithm as in (4.2.6), he used the least mean squares (LMS) algorithm (Widrow and Hoff, 1960), which even though simple, suffers from problems in convergence, especially when the input covariance matrix has a large eigenvalue spread (see Haykin, 1986 for a detailed discussion). Sinha et al. (1979) and Sinha et al, (1981) give some applications in state estimation. The adaptive noise cancellation problem is recast as a system identification problem by Friedlander (1982a) who arrives at a more general infinite impulse response (IIR) adaptive filter structure (cf. the finite impulse response (FIR) structure due to the transversal filter model used in all other development of adaptive noise cancellation theory including ours) which is very complicated and can have stability problems. In several publications, Gardner (1981a) proposed structures for combating the problem of signal leakage into reference input and methods to realize IIR filter structures (Gardner, 1981b).

The least squares solution as in (4.2.6) can be implemented in many different ways. One method to solve for the tap weight vector is by recursive least squares method where matrix inversion is avoided using matrix inversion lemma (Ljung and Soderstrom, 1983). This adaptive method gives exact least squares solutions at every instant of time. Another method is to replace the transversal filter structure by a lattice configuration which is the subject matter of section 4.3. Over and above the advantages provided by the lattice structure in terms of modularity, availability of estimated signal from lower orders of the lattice simultaneously, etc. (discussed in detail in the next section), successive orthogonalization provided only by lattice structures offers advantages in convergence rates. Two types of lattice algorithms are available (i) gradient type and (ii) exact. The gradient type lattice algorithms (Itakura and Saito, 1971; Makhoul, 1975a) have fixed step size and this affects their convergence rate adversely when for example the input is non-Gaussian or has ill-conditioned covariances (Lee, 1980). The exact lattice, however, has a data dependent variable which gives it very rapid convergence rates (Lee, 1980; Haykin, 1986). The name "exact" comes from the fact that, as in the case of recursive least squares mentioned earlier and unlike the case of the gradient type lattice mentioned above, the algorithm gives the exact least squares solution at every instant of time. This implies that parameters, like reflection coefficients of the lattice are not approximated (Itakura and Saito, 1971; Gibson, 1982) but exact expressions are obtained for their update (Goodwin and Sin, 1984). Both the gradient type and exact least squares lattices have found many applications in adaptive noise cancellation (Ayala, 1982; Madhavan et al., 1984; Gardiner et al., 1985). Other than noise cancellation, the joint estimation form can be utilized for adaptive channel equalization (Satorius and Pack, 1981), echo cancellation (Gritton and Lin, 1984), adaptive line enhancement (Reddy et al., 1981), etc.

4.3 Derivation of Weighted Exact Least Squares Lattice Algorithm in the Joint Estimation Form

4.3.1 Introduction

Since the publication of the derivation of exact least squares lattice (LSL) by Morf in 1977, there has been a tremendous amount of interest and published literature in the field. This is not at all surprising since the exact LSL algorithm exhibits modularity in its structure which makes it well-suited for VLSI implementation, has nice stability properties, robustness to variations in the eigenvalue spread of the covariance matrix of input data, successive orthogonalization and decoupling of stages, excellent convergence behaviour and very fast parameter tracking capability (Lee et al., 1981). Alternate derivations of exact LSL algorithm have been presented by Pack and Satorius (1979), Shensa (1981), Lee et al. (1981), Samson (1982), Shichor (1982) and Haykin (1986). In his Ph.D. thesis, Lee (1980) gives both algebraic and geometric derivations of the exact LSL. A comprehensive review of the literature relating to exact LSL algorithm is presented in Friedlander's review paper in 1982(b). Some recent papers have discussed the error propagation properties of LSL algorithms (Ljung and Ljung, 1985; Swanson and Symons, 1984).

In our initial application of adaptive noise cancellation, we implemented the normalized exact least squares lattice algorithm as developed by Lee et al. (1981). This algorithm was tested using both simulated and real evoked potential data (Madhavan et al., 1984). Although, this implementation proved successful in general, we encountered situations where the algorithm had to be resilient to bad or abnormal data. In these cases, the normalized exact LSL algorithm, rather than being resilient, would adapt very rapidly to the bad data (a tribute to the algorithm's fast parameter tracking capability!). Although, fast parameter tracking capability is desirable in general, algorithm's immunity to bad data (robustness) is also important. In an initial attempt to incorporate this feature (discussed in detail in chapter 5, which presents the implementation and testing results for the normalized exact LSL algorithm), we had to use an unconventional weighting scheme involving the forgetting factor, λ.

The above mentioned factor provided the main motivation for developing a lattice algorithm where some suitable weighting scheme can be used for robustness. A second motivation is to have a counterpart to the weighted least squares algorithm (eg., Goodwin and Payne, 1977; Sinha and Kuszta, 1983; Ljung and Soderstrom, 1983) in the area of lattice algorithms, thus giving a very general algorithm which we will call Weighted exact Least Squares Lattice (WLSL) algorithm. The weighting on a performance criterion (eg., equation 4.2.1) can be chosen to give estimates different properties. Using a weighting of the form $g(n,i)$ such that $0 \le g(n,i) \le 1$, where at time n, index i varies from 1 to n, the properties we are interested in are (i) forgetting profile such that statistical variations in observed data can be followed and (ii) robustness such that estimates are resilient to bad or abnormal data.

If $g(n,i) = \lambda^{n-i} \alpha_i$, where $0 \le \lambda, \alpha_i \le 1$, objectives (i) and (ii) mentioned above can be satisfied. Value of λ is chosen according to the expected non-stationarity in the data. The method to choose α_i for robustness will be discussed after the WLSL algorithm is derived. In the development of the WLSL algorithm, we follow closely the lucid and systematic derivation in Haykin (1986).

In deriving the weighted exact least squares lattice algorithm in joint estimation form, we will first derive prediction lattice and then the joint estimation form. Putting the least squares solution discussed in section 4.2 in a general context, it may be noted that (see figure 4.2) d(i) can be any desired signal and u(i), any input signal. Depending on their choices, error, e(i) may take well-known forms. For example, if the desired signal, d(i) is equal to the input, u(i) and input vector $\underline{u}$ are M successively delayed (past) inputs, error e(i) is the forward prediction error (Haykin, 1986). In an analogous fashion, backward prediction error can be obtained.

4.3.2 Adaptive Forward Linear Prediction

The problem of linear prediction has been discussed at length in the literature (eg., Makhoul, 1975b; Burg, 1968; Silvia and Robinson, 1979). Based on the least squares estimation procedure described in section 4.2, let us develop the adaptive forward prediction error filter.

Considering the desired signal as u(i) and input vector $\underline{u}_M(i-1)$ as the M past inputs, i.e.,

$$\underline{u}_M(i-1) = \begin{bmatrix} u(i-1) \\ u(i-2) \\ - \\ - \\ u(i-M) \end{bmatrix}$$

and the tap-weight vector for prediction $\underline{h}(n)$ optimized in the least squares sense over the observation interval $1 \le i \le n$, implying prewindowing (Lee, 1980), Mth order forward prediction error at instant i is

$$f_M(i) = u(i) - \underline{u}_M^T(i-1)\,\underline{h}(n) \tag{4.3.1}$$

where

$$\underline{h}(n) = \begin{bmatrix} h_1(n) \\ - \\ - \\ h_M(n) \end{bmatrix}$$

Equivalently,

$$f_M(i) = \underline{u}^T_{M+1}(i)\,\underline{a}_M(n) \tag{4.3.2}$$

where

$$\underline{u}_{M+1}(i) = \begin{bmatrix} u(i) \\ \underline{u}_M(i-1) \end{bmatrix} \quad \text{and} \quad \underline{a}_M(n) = \begin{bmatrix} 1 \\ -\underline{h}(n) \end{bmatrix}$$

implying a transversal filter structure as shown in figure 4.3 ($f_M(i)$ corresponds to $e_{min}(i)$ in section 4.2).

The performance criterion we choose to minimize in this least squares procedure is a weighted one, where by the use of a weighting of the type $g(n,i) = \lambda^{n-i}\alpha_i$, we can give desired properties to the estimates. In this weighted case, let us denote the minimum value of weighted sum of forward prediction error squares as

$$\mathcal{F}_M(i) = \sum_{i=1}^{n} \lambda^{n-i}\,\alpha_i\,|f_M(i)|^2$$

Analogous to (4.2.8), augmented normal equation for this case can be written as

$$\sum_{i=1}^{n} \lambda^{n-i}\alpha_i \begin{bmatrix} |u(i)|^2 & u(i)\,\underline{u}^T_M(i-1) \\ u(i)\,\underline{u}_M(i-1) & \underline{u}_M(i-1)\underline{u}^T_M(i-1) \end{bmatrix} \begin{bmatrix} 1 \\ -\underline{h}(n) \end{bmatrix} = \begin{bmatrix} \mathcal{F}_M(n) \\ \underline{0}_M \end{bmatrix} \tag{4.3.3}$$

where $\underline{0}_M$ is a null vector of order M.

If we define the matrix on the left hand side as $\Phi_{M+1}(n)$, (4.3.3) can be written as

$$\Phi_{M+1}(n)\,\underline{a}_M(n) = \begin{bmatrix} \mathcal{F}_M(n) \\ \underline{0}_M \end{bmatrix} \tag{4.3.4}$$

$\Phi_{M+1}(n)$ can be seen to be the ((M+1)x(M+1)) deterministic correlation matrix of the input to the forward prediction error filter.

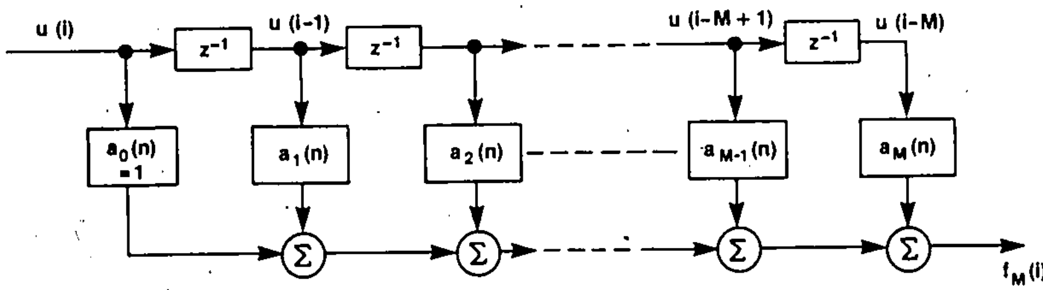

Figure 4.3 Forward prediction error filter of order M.

in figure 4.3.

$$\Phi_{M+1}(n) = \sum_{i=1}^{n} \lambda^{n-i} a_i \underline{u}_{M+1}(i) \underline{u}_{M+1}^T(i)$$

which can be seen to be equivalent to the matrix on the LHS of (4.3.3), keeping in mind the partition

$$\underline{u}_{M+1}(n) = \begin{bmatrix} u(i) \\ \underline{u}_M(i-1) \end{bmatrix}.$$

Define

(i) $$U(n) = \sum_{i=1}^{n} \lambda^{n-i} a_i |u(i)|^2$$

(ii) Deterministic cross-correlation vector between desired response, u(i) and predictor input vector, $\underline{u}_M(i-1)$,

$$\underline{\theta}_1(n) = \sum_{i=1}^{n} \lambda^{n-i} a_i u(i) \underline{u}_M(i-1) \qquad (4.3.5)$$

(iii) (MxM) deterministic correlation matrix of predictor input vector, $\underline{u}_M(i-1)$

$$\Phi_M(n-1) = \sum_{i=1}^{n} \lambda^{n-i} a_i \underline{u}_M(i-1) \underline{u}_M^T(i-1)$$

$$= \sum_{i=1}^{n} \lambda^{(n-1)-i} \beta_i \underline{u}_M(i) \underline{u}_M^T(i) \qquad (4.3.6)$$

$$\text{where } \beta_i = a_{i+1} \qquad (4.3.7)$$

With these three definitions, $\Phi_{M+1}(n)$ can be represented as

$$\Phi_{M+1}(n) = \begin{bmatrix} U(i) & \underline{\theta}_1^T(n) \\ \underline{\theta}_1(n) & \Phi_M(n-1) \end{bmatrix}. \qquad (4.3.8)$$

Separating terms on RHS for i=n in equation (4.3.5) and for i=n-1 in (4.3.6),

$$\underline{\theta}_1(n) = \lambda \underline{\theta}_1(n-1) + a_n u(n) \underline{u}_M(n-1) \qquad (4.3.9)$$

$$\Phi_M(n-1) = \lambda \Phi_M(n-2) + \beta_{n-1} \underline{u}_M(n-1) \underline{u}_M^T(n-1) \qquad (4.3.10)$$

Using the well-known matrix inversion lemma in (4.3.10), where

$$A = \Phi_M(n-1)\,; \quad B^{-1} = \lambda\,\Phi_M(n-2); \quad C = \underline{u}_M(i-1); \quad D = \frac{1}{\beta_{n-1}}$$

and

$$A^{-1} = B - BC(D + C^T + BC)^{-1} C^T B\,,$$

$\Phi_M^{-1}(n-1)$ can be written as

$$\begin{aligned}\Phi_M^{-1}(n-1) &= \lambda^{-1}\Phi_M^{-1}(n-2) - \lambda^{-1}\Phi_M^{-1}(n-2)\,\underline{u}_M(n-1) \\ &\cdot [\beta_{n-1}^{-1} + \underline{u}_M^T(n-1)\lambda^{-1}\Phi_M^{-1}(n-2)\,\underline{u}_M(n-1)]^{-1} \\ &\cdot \underline{u}_M^T(n-1)\lambda^{-1}\Phi_M^{-1}(n-2)\end{aligned} \tag{4.3.11}$$

Define

$$\underline{k}_M'(n-1) = \frac{\lambda^{-1}\Phi_M^{-1}(n-2)\,\underline{u}_M(n-1)}{\frac{1}{\beta_{n-1}} + \lambda^{-1}\underline{u}_M^T(n-1)\,\Phi_M^{-1}(n-2)\,\underline{u}_M(n-1)} \tag{4.3.12}$$

As we shall see presently, the gain vector $\underline{k}'_M(n-1)$ has some interesting interpretations. Using (4.3.12), equation (4.3.11) can be written as

$$\Phi_M^{-1}(n-1) = \lambda^{-1}\Phi_M^{-1}(n-2) - \lambda^{-1}\underline{k}_M'(n-1)\underline{u}_M^T(n-1)\,\Phi_M^{-1}(n-2) \tag{4.3.13}$$

Using (4.3.8) and the expression for $\underline{a}_M(n)$, the augmented normal equation (4.3.4) can be written as

$$\begin{bmatrix} U(n) & \underline{\theta}_1^T(n) \\ \underline{\theta}_1(n) & \Phi_M(n-1) \end{bmatrix} \begin{bmatrix} 1 \\ -\underline{h}(n) \end{bmatrix} = \begin{bmatrix} F_M(n) \\ \underline{0}_M \end{bmatrix} \tag{4.3.14}$$

which is the same as equation (4.3.3).

From (4.3.14)

$$\underline{\theta}_1(n) - \Phi_M(n-1)\,\underline{h}(n) = \underline{0}_M$$

or

$$\underline{h}(n) = \Phi_M^{-1}(n-1)\,\underline{\theta}_1(n) \tag{4.3.15}$$

Using (4.3.9) for $\underline{\theta}_1(n)$, (4.3.13) for $\Phi_M^{-1}(n-1)$ and (4.3.7),

$$\underline{h}(n) = \underline{h}(n-1) + \underline{k}_M'(n-1)[\,u(n) - \underline{u}_M^T(n-1)\,\underline{h}(n-1)] \tag{4.3.16}$$

The term $[u(n) - \underline{u}_M^T(n-1)\ \underline{h}(n-1)]$ has a special significance. Comparing to (4.3.1), one can see that what is different here is that tap-weights from the previous instant, $\underline{h}(n-1)$, are used. Define

$$\eta_M(n) = u(n) - \underline{u}_M^T(n-1)\underline{h}(n-1) = \underline{u}_{M-1}^T(n)\ \underline{a}_M(n-1) \tag{4.3.17}$$

$\eta_M(n)$ is a tentative estimate of forward prediction error $f_M(n)$ before the tap-weight vector is updated. $\eta_M(n)$ is called the forward prediction innovation of order M. Now, (4.3.16) can be written as

$$\underline{h}(n) = \underline{h}(n-1) + \underline{k}_M'(n-1)\eta_M(n) \tag{4.3.18}$$

To find the update equation for forward prediction error filter tap weights, $\underline{a}_M(n)$, write $\underline{a}_M(n)$ as (from (4.3.2))

$$\underline{a}_M(n) = \begin{bmatrix} 1 \\ -\underline{h}(n) \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ -\underline{h}(n-1) - \underline{k}_M'(n-1)\eta_M(n) \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ -h(n-1) \end{bmatrix} - \begin{bmatrix} 0 \\ k_M'(n-1) \end{bmatrix}\eta_M(n)$$

$$\therefore\ \underline{a}_M(n) = \underline{a}_M(n-1) - \begin{bmatrix} 0 \\ \underline{k}_M'(n-1) \end{bmatrix}\eta_M(n) \tag{4.3.19}$$

To find the update equation for the minimum value of performance criterion or sum of weighted forward prediction error squares,

$$\mathcal{F}_M(n) = \sum_{i=1}^{n} \lambda^{n-i}\ \alpha_i\,|f_M(i)|^2$$

$$= \sum_{i=1}^{n} \lambda^{n-i}\ \alpha_i[u(i) - \underline{u}_M^T(i-1)\,\underline{h}(n)]^2$$

Simplification will yield

$$F_M(n) = \sum_{i=1}^{n} \lambda^{n-i} \alpha_i |u(i)|^2 - \underline{\theta}_1^T(n)\underline{h}(n)$$

Substituting update equations for $\underline{\theta}_1(n)$ and $\underline{h}(n)$, it can be shown that

$$F_M(n) = \lambda F_M(n-1) + \alpha_n \eta_M(n) f_M(n) \tag{4.3.20}$$

<u>Interpretation for gain vector</u>, $\underline{k}'_M(n-1)$ [forward prediction case]

From equations (4.3.12) and (4.3.13), the gain vector can be written as

$$\frac{1}{\beta_{n-1}} \underline{k}'_M(n-1) = \Phi_M^{-1}(n-1)\underline{u}_M(n-1)$$

But from (4.3.7), $\beta_{n-1} = \alpha_n$.

$$\therefore \frac{1}{\alpha_n} \underline{k}'_M(n-1) = \Phi_M^{-1}(n-1)\underline{u}_M(n-1)$$

or

$$\underline{k}'_M(n-1) = \Phi_M^{-1}(n-1)[\alpha_n \underline{u}_M(n-1)] \tag{4.3.21}$$

Comparing (4.3.21) to the least squares solution for tap-weight as in (4.3.15), $\underline{k}'_M(n-1)$ can be interpreted as the least squares solution for tap-weights where the input vector is $\underline{u}_M(i-1)$ and the deterministic cross-correlation vector, say $\underline{\theta}'(n)$ in this case, is equal to $\alpha_n \underline{u}_M(n-1)$.

The general definition for the deterministic cross-correlation vector between some desired input, $d'(i)$ and input vector, $\underline{u}_M(i-1)$ is

$$\underline{\theta}'(n) = \sum_{i=1}^{n} \lambda^{n-i} \alpha_i d'(i) \underline{u}_M(i-1) \tag{4.3.22}$$

To give the interpretation, we have to know what this desired input, $d'(i)$, has to be. It can be seen that if

$$d'(i) = 0 \qquad \text{for } i = 1,2,\ldots,n-1\,.$$
$$= 1 \qquad \text{for } i=n$$

equation (4.3.22) becomes

$$\underline{\theta}'(n) = \lambda^0 a_n \underline{u}_M(n-1)$$

which is exactly what is required in equation (4.3.21). So, from (4.3.21), the interpretation for the gain vector, $\underline{k}'_M(n-1)$ is that it is the least squares estimates of the tap-weights when the input vector is $\underline{u}_M(i-1)$ and desired response is 1. This is shown in figure 4.4.

Another quantity of interest in this interpretation is the estimation error, $\gamma'_M(n-1)$, shown in figure 4.4. It can be written as

$$\gamma'_M(n-1) = 1 - \underline{u}_M^T(n-1)\underline{k}'_M(n-1)$$

Substituting for $\underline{k}'_M(n-1)$ from (4.3.12), it can be shown that

$$\gamma'_M(n-1) = \frac{1}{1 + a_n \lambda^{-1} \underline{u}_M^T(n-1)\Phi_M^{-1}(n-2)\underline{u}_M(n-1)} \qquad (4.3.23)$$

Since

$$0 \le \lambda, a_i \le 1 \quad \text{and} \quad \underline{u}_M^T(n-1)\Phi_M^{-1}(n-2)\underline{u}_M(n-1) \ge 0\,.$$

$$0 \le \gamma'_M(n-1) \le 1 \quad \text{or} \quad 0 \le \gamma'_M(n) \le 1$$

Comparing $\gamma'_M(n)$ to similar variables in published literature reviewed in section 4.3.1, specifically Haykin (1986), we see that the effect of weighting used here, i.e., a_n, is that $\gamma'_M(n)$ decreases from 1 to 0 more slowly, or in other words, its trajectory lies above those in published literature. Consequence of this fact, as we shall see in later sections, is that the lattice algorithm can be made robust to bad data. Lee (1980) and Haykin (1986) explain its "likelihood ratio" interpretation. Also notice the difference between Lee's

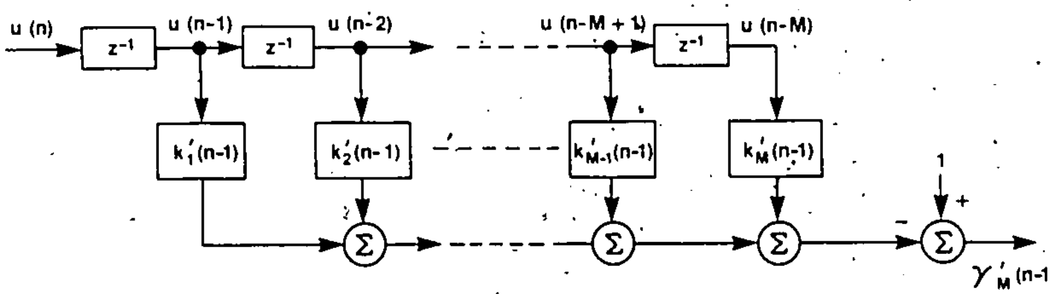

Figure 4.4 Interpretation for gain vector -- forward prediction case.

(1980) definition of "γ" and Haykin's (1986) and our definition. See Haykin (1986) for more detailed discussion of interpretations of "γ".

4.3.3 Adaptive Backward Linear Prediction

Our development here will closely parallel the one we adopted for adaptive forward linear prediction.

Considering the desired signal as u(i-M), M input vector, $\underline{u}_M(i)$, as

$$\underline{u}_M(i) = \begin{bmatrix} u(i) \\ u(i-1) \\ - \\ - \\ u(i-M+1) \end{bmatrix}$$

and the tap-weight vector for backward prediction, g(n) optimized in the least squares sense over the observation interval $1 \le i \le n$, the Mth order backward prediction error at instant, i,

$$b_M(i) = u(i-M) - \underline{u}_M^T(i)\,\underline{g}(n) \tag{4.3.24}$$

where

$$\underline{g}(n) = \begin{bmatrix} g_1(n) \\ - \\ g_M(n) \end{bmatrix}$$

Equivalently,

$$b_M(i) = \underline{u}_{M+1}^T(i)\,\underline{c}_M(n) \tag{4.3.25}$$

where

$$\underline{u}_{M+1}(i) = \begin{bmatrix} \underline{u}_M(i) \\ u(i-M) \end{bmatrix} \text{ and } \underline{c}_M = \begin{bmatrix} -\underline{g}(n) \\ 1 \end{bmatrix}$$

Notice the difference in partitioning used for $\underline{u}_{M+1}(i)$ from the forward prediction case (4.3.2). The transversal filter structure for backward prediction error is shown in figure 4.5. Minimum value of the performance index that is minimized to find the weighted least squares estimate of tap-weights is

$$B_M(n) = \sum_{i=1}^{n} \lambda^{n-i} a_i |b_M(i)|^2$$

Writing the augmented normal equations in a fashion similar to the forward prediction case,

$$\sum_{i=1}^{n} \lambda^{n-i} a_i \begin{bmatrix} \underline{u}_M(i)\,\underline{u}_M^T(i) & u(i-M)\,\underline{u}_M(i) \\ u(i-M)\,\underline{u}_M^T(i) & |u(i-M)|^2 \end{bmatrix} \begin{bmatrix} -\underline{g}(n) \\ \cdot \\ 1 \end{bmatrix} = \begin{bmatrix} \underline{0}_M \\ \cdot \\ B_M(n) \end{bmatrix} \qquad (4.3.26)$$

Matrix on the LHS is

$$\Phi_{M+1}(n) = \sum_{i=1}^{n} \lambda^{n-i} a_i\, \underline{u}_{M+1}(i)\, \underline{u}_{M+1}^T(i)$$

where the partition

$$\underline{u}_{M+1}(i) = \begin{bmatrix} \underline{u}_M(i) \\ u(i-M) \end{bmatrix}$$

is used.

$$\therefore \quad \Phi_{M+1}(n)\, c_M(n) = \begin{bmatrix} \underline{0}_M(i) \\ B_M(n) \end{bmatrix} \qquad (4.3.27)$$

Define

(i) $\mathcal{U}(n-M) = \sum_{i=1}^{n} \lambda^{n-i} a_i |u(i-M)|^2$

(ii) Deterministic cross-correlation vector between desired response, $u(i-M)$ and input vector, $\underline{u}_M(i)$.

$$\underline{\theta}_2(n) = \sum_{i=1}^{n} \lambda^{n-i} a_i\, u(i-M)\, \underline{u}_M(i) \qquad (4.3.28)$$

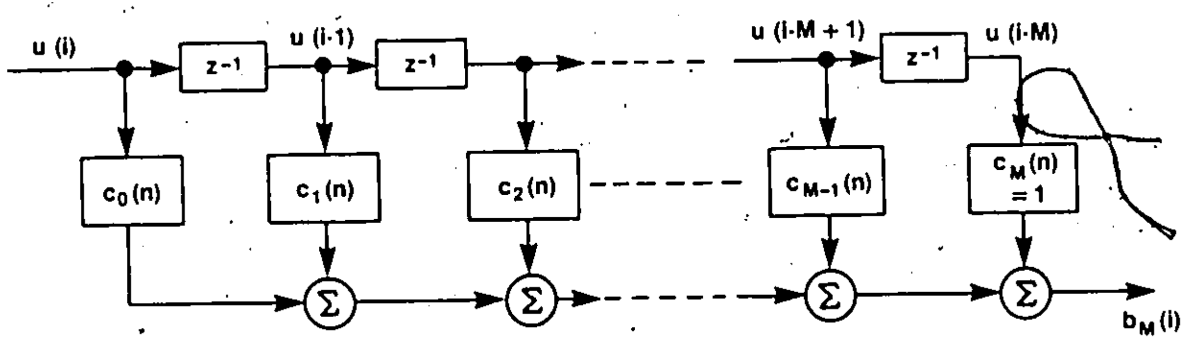

Figure 4.5 Backward prediction error filter of order M.

(iii) (MxM) deterministic correlation matrix of input vector, $\underline{u}_M(i)$

$$\Phi_M(n) = \sum_{i=1}^{n} \lambda^{n-i} \alpha_i \, \underline{u}_M(i) \underline{u}_M^T(i)$$

Then, (4.3.27) can be written as

$$\begin{bmatrix} \Phi_M(n) & \underline{\theta}_2(n) \\ \underline{\theta}_2^T(n) & U(n-M) \end{bmatrix} \begin{bmatrix} -\underline{g}(n) \\ 1 \end{bmatrix} = \begin{bmatrix} \underline{0}_M \\ B_M(n) \end{bmatrix} \tag{4.3.28}$$

Update equations for $\underline{\theta}_2(n)$ and $\Phi_M(n)$ can be written as

$$\underline{\theta}_2(n) = \lambda \underline{\theta}_2(n-1) + \alpha_n u(n-M) \underline{u}_M(n) \tag{4.3.29}$$

$$\Phi_M(n) = \lambda \Phi_M(n-1) + \alpha_n \underline{u}_M(n) \underline{u}_M^T(n) \tag{4.3.30}$$

Using matrix inversion lemma in (4.3.30) and defining

$$\underline{k}_M(n) = \frac{\lambda^{-1} \Phi_M^{-1}(n-1) \underline{u}_M(n)}{\frac{1}{\alpha_n} + \lambda^{-1} \underline{u}_M^T(n) \Phi_M^{-1}(n-1) \underline{u}_M(n)} \tag{4.3.31}$$

we can write

$$\Phi_M^{-1}(n) = \lambda^{-1}\Phi_M^{-1}(n-1) - \lambda^{-1} \underline{k}_M(n) \underline{u}_M^T(n) \Phi_M^{-1}(n-1) \tag{4.3.32}$$

Note that the gain vector, $\underline{k}_M(n)$ in equation (4.3.31) is different from gain vector for the forward prediction case, $\underline{k}'_M(n-1)$ in equation (4.3.12).

Noting from the augmented normal equation in (4.3.28) that

$$\underline{g}(n) = \Phi_M^{-1}(n-1)\underline{\theta}_2(n)$$

and following steps similar to the ones in the forward prediction case, it can be shown that

$$\underline{g}(n) = \underline{g}(n-1) + \underline{k}_M(n)[u(n-M) - \underline{u}_M^T(n)\underline{g}(n-1)]$$

The backward prediction innovation of order M can be defined as

$$\psi_M(n) = u(n-M) - \underline{u}_M^T(n)\underline{g}(n-1) \tag{4.3.32}$$

$$\underline{g}(n) = \underline{g}(n-1) + \underline{k}_M(n)\psi_M(n) \tag{4.3.33}$$

Update of backward prediction error filter tap weights can be written as

$$\underline{c}_M(n) = \begin{bmatrix} -\underline{g}(n) \\ 1 \end{bmatrix} = \begin{bmatrix} -\underline{g}(n-1) - \underline{k}_M(n)\,\psi_M(n) \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} -\underline{g}(n-1) \\ 1 \end{bmatrix} - \begin{bmatrix} \underline{k}_M(n) \\ 0 \end{bmatrix} \psi_M(n) \qquad (4.3.34)$$

As in the case of update equation for $F_M(n)$ in the forward prediction case, sum of weighted backward prediction error squares can be updated as

$$B_M(n) = \lambda B_M(n-1) + \alpha_n \psi_M(n) b_M(n) \qquad (4.3.35)$$

<u>Interpretation for gain vector, $\underline{k}_M(n)$ [backward prediction case]</u>

From equations (4.3.31) and (4.3.32),

$$\underline{k}_M(n) = \Phi_M^{-1}(n)\,\alpha_n \underline{u}_M(n) \qquad (4.3.36)$$

Comparing to gain vector in the forward prediction case as in equation (4.3.21), $\underline{k}_M(n)$ can be interpreted as the least squares estimate of the tap-weights where the input vector is $\underline{u}_M(i)$ and the cross-correlation vector is $\alpha_n\ \underline{u}_M(n)$ implying that the desired signal = 1 for i=n and 0, otherwise. The corresponding transversal filter is shown in figure 4.6. Estimation error, $\gamma_M(n)$ can be written as

$$\gamma_M(n) = 1 - \underline{u}_M^T(n)\,\underline{k}_M(n)$$

$$= \frac{1}{1 + \alpha_n \lambda^{-1} \underline{u}_M^T(n)\,\Phi_M^{-1}(n-1)\underline{u}_M(n)} \qquad (4.3.37)$$

Even though this is different from $\gamma'_M(n)$ in the forward prediction case (4.3.23), $\gamma_M(n)$ has the same range and similar interpretations.

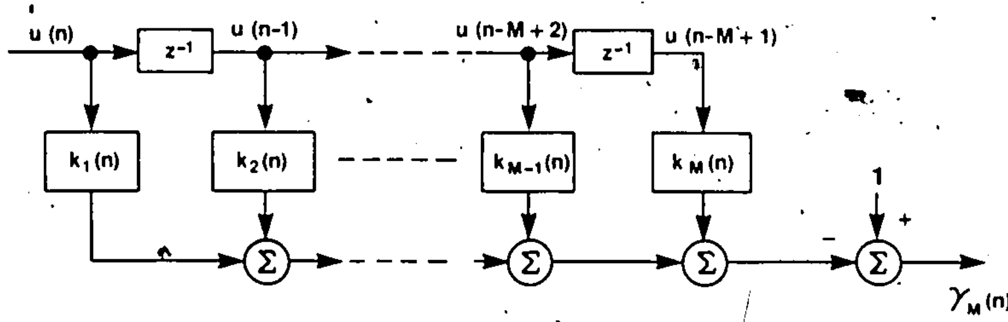

Figure 4.6 Interpretation for gain vector - backward prediction case.

Gain vector update

Before we develop the update equation for gain vector, let us establish the relationships between $\underline{k}$ and γ in the case of backward prediction and corresponding primed quantities in the forward prediction case. Re-writing (4.3.21) and (4.3.23) in the forward prediction case,

$$\frac{1}{\alpha_n}\underline{k}'_M(n-1) = \Phi_M^{-1}(n-1)\,\underline{u}_M(n-1) \tag{4.3.38}$$

$$\gamma'_M(n-1) = 1 - \underline{u}_M^T(n-1)\,\underline{k}'_M(n-1) \tag{4.3.39}$$

The equations (4.3.36) and (4.3.37) in the backward prediction case are

$$\frac{1}{\alpha_n}\underline{k}_M(n) = \Phi_M^{-1}(n)\,\underline{u}_M(n) \tag{4.3.40}$$

$$\gamma_M(n) = 1 - \underline{u}_M^T(n)\,\underline{k}_M(n) \tag{4.3.41}$$

Rewriting (4.3.40) for time instant, (n-1) and comparing to (4.3.38), we see that

$$\underline{k}'_M(n-1) = \frac{\alpha_n}{\alpha_{n-1}}\underline{k}_M(n-1) \tag{4.3.42}$$

Rewriting (4.3.41) for time instant, (n-1), and substituting for $\underline{k}'_M(n-1)$ in (4.3.39), we get

$$\gamma'_M(n-1) = 1 - \frac{\alpha_n}{\alpha_{n-1}} + \frac{\alpha_n}{\alpha_{n-1}}\gamma_M\,(n-1) \tag{4.3.43}$$

Equations (4.3.42) and (4.3.43) gives the relationship between $\underline{k}$ and γ and the primed quantities. Now, we will develop update equations for the gain vector.

Rewriting (4.3.40) for order M+1,

$$\underline{k}_{M+1}(n) = \alpha_n \Phi^{-1}_{M+1}(n)\, \underline{u}_{M+1}(n) \tag{4.3.44}$$

Using the partitioning of $\underline{u}_{M+1}(n)$ in (4.3.2) and $\Phi_{M+1}(n)$ in (4.3.8) and finding $\Phi_{M+1}^{-1}(n)$ by the method of inverse by partitioning,

$$\Phi^{-1}_{M+1}(n) = \begin{bmatrix} 0 & \underline{0}^T_M \\ \underline{0}_M & \Phi^{-1}_M(n-1) \end{bmatrix} + \frac{1}{F_M(n)}\, \underline{a}_M(n)\, \underline{a}^T_M(n)$$

$$\therefore\ \underline{k}_{M+1}(n) = \begin{bmatrix} 0 \\ \alpha_n \Phi^{-1}_M(n-1)\, \underline{u}_M(n-1) \end{bmatrix} + \alpha_n \underline{a}_M(n)\, \frac{f_M(n)}{B_M(n)}$$

$$= \begin{bmatrix} 0 \\ \underline{k}_M(n-1) \end{bmatrix} + \alpha_n \frac{f_M(n)}{F_M(n)}\, \underline{a}_M(n) \tag{4.3.45}$$

using (4.3.39).

The substitution that we did in (4.3.44) can be done from the backward prediction case (equations (4.3.25) and (4.3.28)) to give

$$\Phi^{-1}_{M+1}(n) = \begin{bmatrix} \Phi^{-1}_M(n) & \underline{0}_M \\ \underline{0}^T_M & 0 \end{bmatrix} + \frac{1}{B_M(n)}\, \underline{c}_M(n)\, \underline{c}^T_M(n)$$

and finally

$$\underline{k}_{M+1}(n) = \begin{bmatrix} \alpha_n \Phi^{-1}_M(n)\, \underline{u}_M(n) \\ 0 \end{bmatrix} + \alpha_n \frac{b_M(n)}{B_M(n)}\, \underline{c}_M(n)$$

Recognizing that $\alpha_n \Phi_M^{-1}(n)\, \underline{u}_M(n) = \underline{k}_M(n)$ as in equation (4.3.40),

$$\underline{k}_{M+1}(n) = \begin{bmatrix} \underline{k}_M(n) \\ 0 \end{bmatrix} + \alpha_n \frac{b_M(n)}{B_M(n)} \underline{c}_M(n) \tag{4.3.46}$$

4.3.4 Weighted exact Least Squares Lattice (WLSL)

The results that we developed in the forward and backward prediction and gain update sections can now be combined to produce the WLSL algorithm.

In the backward prediction case, we partition $\underline{u}_{M+1}(i)$ as

$$\underline{u}_{M+1}(i) = \begin{bmatrix} \underline{u}_M(i) \\ u(i-M) \end{bmatrix}$$

and it results in the correlation matrix, as in equation (4.3.28),

$$\Phi_{M+1}(n) = \begin{bmatrix} \Phi_M(n) & \underline{\theta}_2(n) \\ \underline{\theta}_2^T(n) & U(n-M) \end{bmatrix}$$

Multiply both sides of the above equation by a (M+1) vector whose first M elements are $\underline{a}_{M-1}(n)$ and last element, zero.

$$\Phi_{M+1}(n) \begin{bmatrix} \underline{a}_{M-1}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} \Phi_M(n) & \underline{a}_{M-1}(n) \\ \underline{\theta}_2^T(n) & \underline{a}_{M-1}(n) \end{bmatrix}$$

From augmented normal equation in the forward prediction case (4.3.4), we know

$$\Phi_M(n)\underline{a}_{M-1}(n) = \begin{bmatrix} F_{M-1}(n) \\ \underline{0}_{M-1} \end{bmatrix}$$

Also, define a new quantity $\Delta_{M-1}(n) = \underline{\theta}_2^T(n)\, \underline{a}_{M-1}(n)$. Using these two expressions in (4.3.47) yields

$$\Phi_{M+1}(n)\begin{bmatrix} \underline{a}_{M-1}(n) \\ 0 \end{bmatrix} = \begin{bmatrix} F_{M-1}(n) \\ \underline{0}_{M-1} \\ \Delta_{M-1}(n) \end{bmatrix} \qquad (4.4.48)$$

Similarly, approaching from the forward prediction case, the equation for correlation matrix (equation (4.3.8)),

$$\Phi_{M+1}(n) = \begin{bmatrix} U(n) & \underline{\theta}_1^T(n) \\ \underline{\theta}_1(n) & \Phi_M(n-1) \end{bmatrix}$$

Multiply both sides by an (M+1) vector whose first element is zero and last M elements are $\underline{c}_{M-1}(n-1)$.

$$\Phi_{M+1}(n)\begin{bmatrix} 0 \\ \underline{c}_{M-1}(n-1) \end{bmatrix} = \begin{bmatrix} \underline{\theta}_1^T(n) & \underline{c}_{M-1}(n) \\ \Phi_M(n-1) & \underline{c}_{M-1}(n-1) \end{bmatrix} \qquad (4.3.49)$$

From the augmented normal equation in the backward prediction case (4.3.27),

$$\Phi_M(n-1)\underline{c}_{M-1}(n-1) = \begin{bmatrix} \underline{0}_{M-1} \\ B_{M-1}(n-1) \end{bmatrix}$$

Define $\Delta'_{M-1}(n) = \underline{\theta}_1^T(n)\ \underline{c}_{M-1}(n-1)$

Substituting these expressions in (4.3.49),

$$\Phi_{M+1}(n)\begin{bmatrix} 0 \\ \underline{c}_{M-1}(n-1) \end{bmatrix} = \begin{bmatrix} \Delta'_{M-1}(n) \\ \underline{0}_{M-1} \\ B_{M-1}(n-1) \end{bmatrix}. \qquad (4.3.50)$$

Multiplying both sides by

$$\frac{\Delta_{M-1}(n)}{B_{M-1}(n-1)}$$

$$\Phi_{M+1}(n)\begin{bmatrix} 0 \\ \underline{c}_{M-1}(n-1) \end{bmatrix} \frac{\Delta_{M-1}(n)}{B_{M-1}(n-1)} = \begin{bmatrix} \dfrac{\Delta_{M-1}(n)\Delta'_{M-1}(n)}{B_{M-1}(n-1)} \\ \underline{0}_{M-1} \\ \Delta_{M-1}(n) \end{bmatrix} \quad (4.3.51)$$

Subtracting (4.3.51) from (4.3.48),

$$\Phi_{M+1}(n)\left[\begin{bmatrix} \underline{a}_{M-1}(n) \\ 0 \end{bmatrix} - \frac{\Delta_{M-1}(n)}{B_{M-1}(n-1)}\begin{bmatrix} 0 \\ \underline{c}_{M-1}(n-1) \end{bmatrix}\right] = \begin{bmatrix} F_{M-1}(n) - \dfrac{\Delta_{M-1}(n)\Delta'_{M-1}(n)}{B_{M-1}(n-1)} \\ \underline{0}_M \end{bmatrix}$$

Comparing this equation to the augmented normal equation in the forward prediction case, which is,

$$\Phi_{M+1}(n)\underline{a}_M(n) = \begin{bmatrix} F_M(n) \\ \underline{0}_{M-1} \end{bmatrix}$$

we get

$$\underline{a}_M(n) = \begin{bmatrix} \underline{a}_{M-1}(n) \\ 0 \end{bmatrix} - \frac{\Delta_{M-1}(n)}{B_{M-1}(n-1)}\begin{bmatrix} 0 \\ \underline{c}_{M-1}(n-1) \end{bmatrix} \quad (4.3.52)$$

$$F_M(n) = F_{M-1}(n) - \frac{\Delta_{M-1}(n)\Delta'_{M-1}(n)}{B_{M-1}(n-1)} \quad (4.3.53)$$

Multiplying (4.3.48) by

$$\frac{\Delta'_{M-1}(n)}{F_{M-1}(n)}$$

and subtracting the resulting expression from (4.3.50), we get

$$\Phi_{M+1}(n)\left[\begin{bmatrix} 0 \\ \underline{c}_{M-1}(n-1) \end{bmatrix} - \frac{\Delta'_{M-1}(n)}{F_{M-1}(n)}\begin{bmatrix} \underline{a}_{M-1}(n) \\ 0 \end{bmatrix}\right] = \begin{bmatrix} \underline{0}_M \\ B_{M-1}(n-1) - \dfrac{\Delta_{M-1}(n)\Delta'_{M-1}(n)}{F_{M-1}(n)} \end{bmatrix}$$

Comparing this to the augmented normal equation in the backward prediction case, which is,

$$\Phi_{M+1}(n)\ \underline{c}_M(n) = \begin{bmatrix} \underline{0}_M \\ B_M(n) \end{bmatrix}$$

we get

$$\underline{c}_M(n) = \begin{bmatrix} 0 \\ \underline{c}_{M-1}(n-1) \end{bmatrix} - \frac{\Delta'_{M-1}(n)}{F_{M-1}(n)} \begin{bmatrix} \underline{a}_{M-1}(n) \\ 0 \end{bmatrix} \qquad (4.3.54)$$

$$B_M(n) = B_{M-1}(n-1) - \frac{\Delta_{M-1}(n)\Delta'_{M-1}(n)}{F_{M-1}(n)} \qquad (4.3.55)$$

To find the relationship between $\Delta_{M-1}(n)$ and $\Delta'_{M-1}(n)$, pre-multiply both sides of (4.3.48) by the vector $[0\ \ \underline{c}^T_{M-1}(n-1)]$. We get

$$[0\ \ \underline{c}^T_{M-1}(n-1)]\ \Phi_{M+1}(n) \begin{bmatrix} \underline{a}_{M-1}(n) \\ 0 \end{bmatrix} = [0\ \ \underline{c}^T_{M-1}(n-1)] \begin{bmatrix} F_{M-1}(n) \\ \underline{0}_{M-1} \\ \Delta_{M-1}(n) \end{bmatrix} \qquad (4.3.56)$$

$$= \Delta_{M-1}(n),$$

since last element of $\underline{c}^T_{M-1}(n-1)$ is unity.

Taking transpose of both sides of (4.3.50),

$$[0\ \ \underline{c}^T_{M-1}(n-1)]\ \Phi_{M+1}(n) = [\Delta'_{M-1}(n)\ \ \underline{0}^T_{M-1}\ \ B_{M-1}(n-1)]$$

Substituting this in the LHS of (4.3.56),

$$[\Delta'_{M-1}(n)\ \ \underline{0}^T_{M-1}\ \ B_{M-1}(n-1)] \begin{bmatrix} \underline{a}_{M-1}(n) \\ 0 \end{bmatrix} = \Delta'_{M-1}(n)$$

since the first element of $\underline{a}_{M-1}(n) = 1$

$$\therefore\ \ \Delta'_{M-1}(n) = \Delta_{M-1}(n)$$

Define forward reflection coefficient as

$$\Gamma_{f,M}(n) = \frac{\Delta_{M-1}(n)}{B_{M-1}(n-1)} \tag{4.3.57}$$

and backward reflection coefficient as

$$\Gamma_{b,M}(n) = \frac{\Delta_{M-1}(n)}{F_{M-1}(n)} = \frac{\Delta_{M-1}(n)}{F_{M-1}(n)} \tag{4.3.58}$$

(See Robinson and Treitel (1980) for an interesting discussion on reflection coefficients.) Using the definitions of reflection coefficients in (4.3.52) and (4.3.54), we get

$$\underline{a}_M(n) = \begin{bmatrix} \underline{a}_{M-1}(n) \\ 0 \end{bmatrix} - \Gamma_{f,M} \begin{bmatrix} 0 \\ \underline{c}_{M-1}(n-1) \end{bmatrix} \tag{4.3.59}$$

and

$$\underline{c}_M(n) = \begin{bmatrix} 0 \\ \underline{c}_{M-1}(n-1) \end{bmatrix} - \Gamma_{b,M} \begin{bmatrix} \underline{a}_{M-1}(n) \\ 0 \end{bmatrix} \tag{4.3.60}$$

Multiplying LHS of (4.3.59) by $\underline{u}^T_{M+1}(n)$, the first term on the RHS by

$$\underline{u}^T_{M+1}(n) = \begin{bmatrix} \underline{u}_M(n) \\ u(n-M) \end{bmatrix}^T$$

and the second term by

$$\underline{u}^T_{M+1}(n) = \begin{bmatrix} u(n) \\ \underline{u}_M(n-1) \end{bmatrix}^T$$

we get

$$\underline{u}^T_{M+1}(n)\,\underline{a}_M(n) = \underline{u}^T_M(n)\,\underline{a}_{M-1}(n) - \Gamma_{f,M}(n)\,b_{M-1}(n-1)$$

or

$$f_M(n) = f_{M-1}(n) - \Gamma_{f,M}(n)\,b_{M-1}(n-1) \tag{4.3.61}$$

A similar procedure with (4.3.60) will yield

$$b_M(n) = b_{M-1}(n-1) - \Gamma_{b,M}(n)\,f_{M-1}(n) \tag{4.3.62}$$

Equations (4.3.61) and (4.3.62) lead to the lattice structure shown in figure 4.7. Two more update equations for "γ" and "Δ" are required to complete the WLSL algorithm. In developing this, we have to keep in mind that in the case of WLSL, the "γ"'s for forward and backward predictions are distinct (as opposed to in the unweighted LSL case) and are related by (4.3.43),

$$\gamma'_M(n-1) = 1 - \frac{\alpha_n}{\alpha_{n-1}} + \frac{\alpha_n}{\alpha_{n-1}}\,\gamma_M(n-1) \qquad (4.3.63)$$

Rewriting equation (4.3.46) for order, M, and time, (n-1),

$$\underline{k}_M(n-1) = \begin{bmatrix} \underline{k}_{M-1}(n-1) \\ 0 \end{bmatrix} + \alpha_{n-1}\frac{b_{M-1}(n-1)}{B_{M-1}(n-1)}\,\underline{c}_{M-1}(n-1)$$

Multiplying both sides by $\underline{u}^T_M(n-1)$,

$$\underline{u}^T_M(n-1)\,\underline{k}_M(n-1) = \underline{u}^T_M(n-1)\begin{bmatrix} \underline{k}_{M-1}(n-1) \\ 0 \end{bmatrix} +$$

$$\alpha_{n-1}\frac{b_{M-1}(n-1)}{B_{M-1}(n-1)}\,\underline{u}^T_M(n-1)\,\underline{c}_{M-1}(n-1) \qquad (4.3.64)$$

Equation (4.3.41) is

$$\gamma_M(n) = 1 - \underline{u}^T_M(n)\,\underline{k}_M(n)$$

Now it can be identified that LHS of (4.3.64) is $1-\gamma'_M(n-1)$.

First term on RHS of (4.3.64) is

$$[\underline{u}^T_{M-1}(n-1)\;\; u(n-M)]\begin{bmatrix} \underline{k}_{M-1}(n-1) \\ 0 \end{bmatrix} = \underline{u}^T_{M-1}(n-1)\,\underline{k}_{M-1}(n-1)$$

$$= 1 - \gamma_{M-1}(n-1)$$

In the second term on RHS of (4.3.64),

$$\underline{u}^T_M(n-1)\,\underline{c}_{M-1}(n-1) = b_{M-1}(n-1)$$

Therefore (4.3.64) can be written as

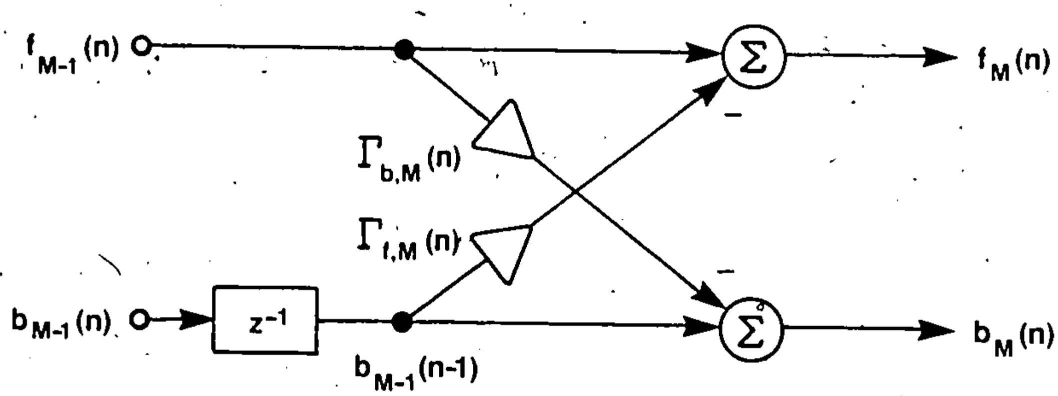

Figure 4.7 Lattice structure.

$$\gamma_M(n-1) = \gamma_{M-1}(n-1) - a_{n-1}\frac{|b_{M-1}(n-1)|^2}{B_{M-1}(n-1)} \qquad (4.3.65)$$

Writing for time = n, (4.3.65) is

$$\gamma_M(n) = \gamma_{M-1}(n) - a_n\frac{|b_{M-1}(n)|^2}{B_{M-1}(n)} \qquad (4.3.66)$$

Knowing the relationship between $\gamma_M(n-1)$ and $\gamma'_M(n-1)$ as in equation (4.3.63), (4.3.65) can be written for $\gamma'_M(n-1)$ by substituting

$$\gamma_M(n-1) = 1 - \frac{a_{n-1}}{a_n} + \frac{a_{n-1}}{a_n}\gamma'_M(n-1)$$

The result is the update equation for $\gamma_M'(n-1)$ and it is

$$\gamma'_M(n-1) = \gamma'_{M-1}(n-1) - a_n\frac{|b_{M-1}(n-1)|^2}{B_{M-1}(n-1)} \qquad (4.3.67)$$

A certain amount of non-intuitive algebraic manipulations are required to find update equations for $\Delta_{M-1}(n)$. Let us start by noting that since the first element of the M vector, $\underline{a}_{M-1}(n-1)$ is unity, $\Delta_{M-1}(n)$ can be written as

$$\Delta_{M-1}(n) = [\Delta_{M-1}(n)\ \underline{0}^T_{M-1}\ B_{M-1}(n-1)]\begin{bmatrix}\underline{a}_{M-1}(n-1)\\ 0\end{bmatrix}$$

Substituting for $[\Delta_{M-1}(n)\ \underline{0}^T_{M-1}\ B_{M-1}(n-1)]$ from (4.3.50), the above equation can be written as

$$\Delta_{M-1}(n) = [0\ \underline{c}^T_{M-1}(n-1)]\,\Phi_{M+1}(n)\begin{bmatrix}\underline{a}_{M-1}(n-1)\\ 0\end{bmatrix}$$

Further substitution for $\Phi_{M+1}(n)$ as

$$\Phi_{M+1}(n) = \lambda\,\Phi_{M+1}(n-1) + a_n\,\underline{u}_{M+1}(n)\,\underline{u}^T_{M+1}(n)$$

yields

$$\Delta_{M-1}(n) = \lambda [0 \ \underline{c}^T_{M-1}(n-1)] \Phi_{M+1}(n-1) \begin{bmatrix} \underline{a}_{M-1}(n-1) \\ 0 \end{bmatrix}$$

$$+ \alpha_n [0 \ \underline{c}^T_{M-1}(n-1)] \underline{u}_{M+1}(n) \underline{u}^T_{M+1}(n) \begin{bmatrix} \underline{a}_{M-1}(n-1) \\ 0 \end{bmatrix} \qquad (4.3.68)$$

Several substitutions can be performed in this equation. From (4.3.48),

$$\Phi_{M+1}(n-1) \begin{bmatrix} \underline{a}_{M-1}(n-1) \\ 0 \end{bmatrix} = \begin{bmatrix} F_{M-1}(n-1) \\ \underline{0}_{M-1} \\ \Delta_{M-1}(n-1) \end{bmatrix}.$$

and since last element of $\underline{c}_{M-1}(n-1)$ is unity, first term on RHS of (4.3.68) can be written as $\lambda\Delta_{M-1}(n-1)$.

Using the partition

$$\underline{u}_{M+1}(n) = \begin{bmatrix} \underline{u}_M(n) \\ u(n-M) \end{bmatrix},$$

$$\underline{u}^T_{M+1}(n) \begin{bmatrix} \underline{a}_{M-1}(n-1) \\ 0 \end{bmatrix} = \eta_{M-1}(n)$$

and from the other partition,

$$\underline{u}_{M+1}(n) = \begin{bmatrix} u(n) \\ \underline{u}_M(n-1) \end{bmatrix},$$

$$[0 \ \underline{c}^T_{M-1}(n-1)] \underline{u}_{M+1}(n) = b_{M-1}(n-1)$$

Using all these simplifications in (4.3.68),

$$\Delta_{M-1}(n) = \lambda \Delta_{M-1}(n-1) + \alpha_n b_{M-1}(n-1) \eta_{M-1}(n) \qquad (4.3.69)$$

Starting with equation (4.3.19) and multiplying both sides by $\underline{u}^T_{M+1}(n)$, it can be shown that

$$f_M(n) = \eta_M(n)\,\gamma'_M(n-1)$$

For order (M-1), $\eta_{M-1}(n)$ can be written in two ways using the relationship between $\gamma'_{M-1}(n-1)$ and $\gamma_{M-1}(n-1)$ in (4.3.63).

$$\eta_{M-1}(n) = \frac{f_{M-1}(n)}{\gamma'_{M-1}(n-1)} \tag{4.3.70}$$

$$= \frac{f_{M-1}(n)}{1 - \dfrac{\alpha_n}{\alpha_{n-1}} + \dfrac{\alpha_n}{\alpha_{n-1}}\,\gamma_{M-1}(n-1)} \tag{4.3.71}$$

Substituting (4.3.70) in (4.3.69),

$$\Delta_{M-1}(n) = \lambda\,\Delta_{M-1}(n-1) + \alpha_n\,\frac{b_{M-1}(n-1)\,f_{M-1}(n)}{\gamma'_{M-1}(n-1)} \tag{4.3.72}$$

Substituting (4.3.71) in (4.3.69),

$$\Delta_{M-1}(n) = \lambda\,\Delta_{M-1}(n-1) + \frac{b_{M-1}(n-1)\,f_{M-1}(n)}{\dfrac{1}{\alpha_n} - \dfrac{1}{\alpha_{n-1}} + \dfrac{\gamma_{M-1}(n-1)}{\alpha_{n-1}}} \tag{4.3.73}$$

Equations (4.3.72), (4.3.57), (4.3.58), (4.3.61), (4.3.62), (4.3.53), (4.3.55) and (4.3.67) form the Weighted exact Least Squares Lattice algorithm.

(i) $$\Delta_{M-1}(n) = \lambda\,\Delta_{M-1}(n-1) + \alpha_n\,\frac{b_{M-1}(n-1)\,f_{M-1}(n)}{\gamma_{M-1}(n-1)}$$

(ii) $$\Gamma_{f,M}(n) = \frac{\Delta_{M-1}(n)}{B_{M-1}(n-1)}$$

(iii) $$\Gamma_{b,M}(n) = \frac{\Delta_{M-1}(n)}{F_{M-1}(n)}$$

(iv) $$f_M(n) = f_{M-1}(n) - \Gamma_{f,M}(n)\,b_{M-1}(n-1)$$

(v) $$b_M(n) = b_{M-1}(n-1) - \Gamma_{b,M}(n)\, f_{M-1}(n)$$

(vi) $$F_M(n) = F_{M-1}(n) - \frac{|\Delta_{M-1}(n)|^2}{B_{M-1}(n-1)}$$

(vii) $$B_M(n) = B_{M-1}(n-1) - \frac{|\Delta_{M-1}(n)|^2}{F_{M-1}(n)}.$$

(viii) $$\gamma_M(n-1) = \gamma_{M-1}(n-1) - \alpha_n \frac{|b_{M-1}(n-1)|^2}{B_{M-1}(n-1)}$$

It may be noted that instead of (i) and (viii), equations (4.3.73) and (4.3.66) could be used in the WLSL algorithm.

(i)' $$\Delta_{M-1}(n) = \lambda\, \Delta_{M-1}(n-1) - \frac{b_{M-1}(n-1)\, f_{M-1}(n)}{\frac{1}{\alpha_n} - \frac{1}{\alpha_{n-1}} + \frac{\gamma_{M-1}(n-1)}{\alpha_{n-1}}}$$

(viii)' $$\gamma_M(n) = \gamma_{M-1}(n) - \alpha_n \frac{|b_{M-1}(n)|^2}{B_{M-1}(n)}$$

A few comments are in order regarding the WLSL update equations above.

Remark 1. When the weighting $\alpha_i=1$, equations (i) to (viii) including (i)' and (viii)' reduce to unweighted LSL (cf. Haykin (1986)). When $\alpha_i=1$, $\gamma_M(n) = \gamma'_M(n)$ (see equation (4.3.63).

Remark 2. When α_i is greater than or equal to zero but less than unity, $\gamma'_{M-1}(n-1)$ remains larger than in the unweighted case, as was discussed in the gain vector update section. This can also be seen from update equation (viii).

To fully appreciate the consequence of this property, let us look at the "likelihood ratio" interpretation of "γ" in Lee (1980) on page 124. Dropping the order subscript and using an "L" superscript to indicate his definition of "γ", which is different from ours,

$$\gamma^L(n) = \underline{u}_M^T(n)\,\Phi_M^{-1}(n)\,\underline{u}_M(n)$$

Dropping the weightings λ and α for the moment, our "γ" is defined as

$$\gamma(n) = \frac{1}{1 + \underline{u}_M^T(n)\,\Phi_M^{-1}(n-1)\,\underline{u}_M(n)} \tag{4.3.74}$$

Lee's "γ" can also be written as (Lee (1980), page 120) -

$$\gamma^L(n) = \frac{\underline{u}_M^T(n)\,\Phi_M^{-1}(n-1)\,\underline{u}_M(n)}{1 + \underline{u}_M^T(n)\,\Phi_M^{-1}(n-1)\,\underline{u}_M(n)} \tag{4.3.75}$$

$$\therefore \gamma(n) = 1 - \gamma^L(n) \tag{4.3.76}$$

The interpretation of likelihood ratio, $\gamma^L(n)$ is that it is a measure of how non-Gaussian the observation is, i.e., when $\gamma^L(n)$ is close to one, the observation is non-Gaussian. In the unweighted case, our $\gamma(n)$ (= $\gamma'(n)$) will then be close to zero. Therefore, our "γ" is a direct measure of how Gaussian the observation is. From update equation (i), it can be seen that the contribution of second term on the RHS will become large and the lattice will adapt rapidly. When there are outliers in the data (or bad data), to avoid this rapid adaptation from occuring, the fact mentioned at the beginning of remark 2. that $\gamma'_{M-1}(n-1)$ remains larger in the weighted case implies that the lattice sees the observation as more Gaussian than it really is and adapts less. Furthermore, the fact that weighting, α_i, is chosen such that it is small when the observation is an outlier further reduces the contribution of the second term in update equation (i).

Remark 3. When $\alpha_i = 0$, the lattice is shut down completely. No adaptation takes place.

Remark 4. One method to choose $\{\alpha_i\}$ is to pre-determine the sequence from prior knowledge about the data being processed.

From the update equations, it is obvious that at time instant, n, α_n has to be available before any updates are done. This may make on-line choice of α_n difficult.

One way to avoid pre-determination of $\{\alpha_i\}$ may be to have a two pass system where during the first pass, $\alpha_i=1$ and fluctuations of $\gamma_{M-1}(n-1)$ are noted so that during the second pass, for those instances where $\gamma_{M-1}(n-1)$ were very small (indicating non-Gaussian observations or outliers), α is made small after the initial adaptation period, thus making the WLSL robust.

4.3.5 Joint Estimation Form

Going back to the problem discussed in section 4.2 of noise cancellation or joint process estimation, we have to estimate a process $\{d(i)\}$ from observations of a related process $\{u(i)\}$. This can be done by adding a transversal filter section to the lattice structure as shown in figure 4.8. This section estimates the value of $d(i)$ at any instant from $\{b_M(i)\}$ using tap weights $\{p_M(i)\}$. It can be shown that $\{b_M(i)\}$ has the same information content as $\{u(i)\}$ and that $b_0(i)$, $b_1(i)$, ... $b_M(i)$ are uncorrelated with each other in a deterministic sense for all instants of time (Haykin, 1986).

From the foregoing discussion, the estimate of $d(i)$, given $u(i)$ is

$$\hat{d}_M(i|u(i)) = \underline{b}_M^T(i)\,\underline{p}_M(i) \tag{4.3.77}$$

where the (M+1) vectors $\underline{b}_M(i)$ and $\underline{p}_M(i)$ are

$$\underline{b}_M(i) = \begin{bmatrix} b_0(i) \\ b_1(i) \\ \cdot \\ \cdot \\ \cdot \\ b_M(i) \end{bmatrix} \qquad \underline{p}_M(i) = \begin{bmatrix} p_0(i) \\ p_1(i) \\ \cdot \\ \cdot \\ \cdot \\ p_M(i) \end{bmatrix}$$

The performance criterion to be minimized in the least squares solution of $\underline{p}_M(i)$ is

$$E_M(n) = \sum_{i=1}^{n} \lambda^{n-i} a_i\,|e_M(i)|^2$$

where

$$e_M(i) = d(i) - \hat{d}_M(i|u(i)) = d(i) - \underline{b}_M^T(i)\,\underline{p}_M(i) \tag{4.3.78}$$

In the spirit of earlier discussions on the least squares solution, the estimates of tap-weights can be written as

$$D_{M+1}(n)\,\underline{p}_M(n) = \underline{t}_M(n) \tag{4.3.79}$$

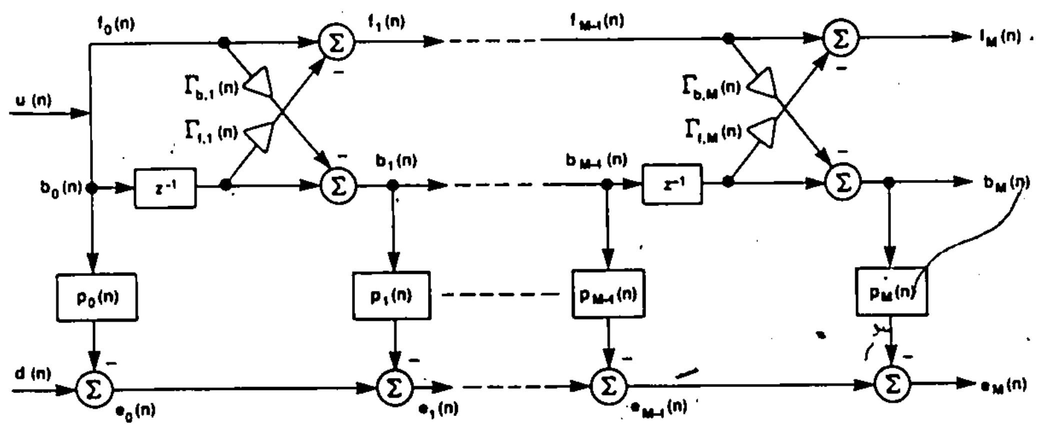

Figure 4.8 Weighted exact least squares lattice for joint process estimation.

The $((M+1)x(M+1))$ correlation matrix of backward prediction errors, $D_{M+1}(n)$ is defined as

$$D_{M+1}(n) = \sum_{i=1}^{n} \lambda^{n-i} \alpha_i \underline{b}_M(i) \underline{b}_M^T(i) \qquad (4.3.80)$$

Backward prediction errors being uncorrelated to each other, it can be shown in a straight-forward manner that

$$D_{M+1}(n) = \begin{bmatrix} B_0(n) & & & \\ & B_1(n) & & 0 \\ & 0 & \ddots & \\ & & & B_M(n) \end{bmatrix} \qquad (4.3.81)$$

The $(M+1)$ cross-correlation vector between $d(i)$ and $\underline{b}_M(i)$ is defined as

$$\underline{t}_M = \sum_{i=1}^{n} \lambda^{n-i} \alpha_i d(i) \underline{b}_M(i) \qquad (4.3.82)$$

Define a lower triangular matrix $L_M(n)$ (of order $((M+1)x(M+1))$) of backward prediction error filter coefficients (see section 4.3.3), where the second subscript indicates the order of the filter as

$$L_M(n) = \begin{bmatrix} 1 & & & 0 \\ c_{0,1}(n) & 1 & & \\ \vdots & & \ddots & \\ c_{0,M}(n) & \cdots & c_{M-1,M}(n) & 1 \end{bmatrix} = \begin{bmatrix} \underline{c}_0^T(n) \\ \underline{c}_1^T(n) \\ \vdots \\ \underline{c}_M^T(n) \end{bmatrix}$$

This, along with the definition of backward prediction error in (4.3.25), let us write (4.3.82) as

$$\underline{t}_M(n) = L_M(n) \sum_{i=1}^{n} \lambda^{n-i} \alpha_i d(i) \underline{u}_{M+1}(i)$$
$$= L_M(n) \underline{\theta}_{M+1}(n) \qquad (4.3.83)$$

where $\underline{\theta}_{M+1}(n)$ is a new (M+1) cross-correlation vector between d(i) and $\underline{u}_{M+1}(i)$.

Now, (4.3.79) yields

$$\underline{p}_M(n) = D^{-1}_{M+1}(n) \begin{bmatrix} \underline{c}_0^T(n) \\ \underline{c}_1^T(n) \\ \cdot \\ \cdot \\ \cdot \\ \underline{c}_M^T(n) \end{bmatrix} \underline{\theta}_{M+1}(n)$$

Since $D_{M+1}(n)$ is diagonal as in (4.3.81) and with some abuse of notation, we can write the equation for the M^{th} tap weight as

$$p_M(n) = B_M^{-1}(n)\ \underline{c}_M^T(n)\ \underline{\theta}_{M+1}(n)$$

Defining the scalar,

$$\rho_M(n) = \underline{c}_M^T(n)\ \underline{\theta}_{M+1}(n), \qquad (4.3.84)$$

$$p_M(n) = \frac{\rho_M(n)}{B_M(n)}, \qquad (4.3.85)$$

To find the update equation for $\rho_M(n)$, let us start with equation (4.3.34),

$$\underline{c}_M(n) = \underline{c}_M(n-1) - \begin{bmatrix} \underline{k}_M(n) \\ 0 \end{bmatrix} \psi_M(n)$$

Multiplying throughout by $\underline{u}^T_{M+1}(n)$ and simplifying, we get

$$b_M(n) = \psi_M(n)[1 - \underline{u}_M^T(n)\underline{k}_M(n)]$$
$$= \psi_M(n)\gamma_M(n), \qquad \text{from (4.3.37)}$$

Therefore

$$\underline{c}_M(n) = \underline{c}_M(n-1) - \frac{b_M(n)}{\gamma_M(n)} \begin{bmatrix} \underline{k}_M(n) \\ 0 \end{bmatrix}$$

Using this and the usual update of $\underline{\theta}_{M+1}(n)$ as

$$\underline{\theta}_{M+1}(n) = \lambda\, \underline{\theta}_{M+1}(n-1) + \alpha_n\, d(n)\, \underline{u}_{M+1}(n)$$

in equation (4.3.84) and simplifying,

$$\rho_M(n) = \lambda\, \rho_M(n-1) + \alpha_n \frac{d(n)\, b_M(n)}{\gamma_M(n)} - \frac{b_M(n)}{\gamma_M(n)} \underline{k}_M^T(n) \underline{\theta}_M(n) \qquad (4.3.86)$$

Substituting for $\underline{k}^T_M(n)$ from (4.3.36), the last term equals

$$\alpha_n \frac{b_M(n)}{\gamma_M(n)} \underline{u}_M^T(n)\, \Phi_M^{-1}(n)\, \underline{\theta}_M(n)$$

Comparing to the usual least squares solution, $\Phi^{-1}{}_M(n)\ \underline{\theta}_M(n)$ is tap-weight estimates when the desired signal is d(i) and input vector is $\underline{u}_M(i)$. So, when input is applied to the tap-weights, $\Phi^{-1}{}_M(n)\ \underline{\theta}_M(n)$, we get an estimate of the desired signal.

$$\therefore\ \underline{u}_M^T(n)\, \Phi_M^{-1}(n)\, \underline{\theta}_M(n) = \hat{d}_{M-1}(n \mid u(n))$$

Substituting this result into the last term of (4.3.86),

$$\rho_M(n) = \lambda\, \rho_M(n-1) + \alpha_n \frac{b_M(n)}{\gamma_M(n)} [d(n) - \hat{d}_{M-1}(n \mid u(n))]$$

Using (4.3.78),

$$\rho_M(n) = \lambda\, \rho_M(n-1) + \alpha_n \frac{b_M(n)}{\gamma_M(n)} e_{M-1}(n) \qquad (4.3.87)$$

Equation (4.3.78) for order (M-1) is

$$e_{M-1}(n) = d(n) - \underline{b}^T_{M-1}(n)\,\underline{p}_{M-1}(n) \tag{4.3.87}$$

$$= d(n) - \sum_{j=0}^{M-1} p_j(n)\,b_j(n)$$

$$= -p_{M-1}(n)\,b_{M-1}(n) + [d(n) - \sum_{j=0}^{M-2} p_j(n)\,b_j(n)]$$

The last term is $e_{M-2}(n)$.

$$\therefore e_{M-1}(n) = -p_{M-1}(n)\,b_{M-1}(n) + e_{M-2}(n)$$

Writing for order M,

$$e_M(n) = e_{M-1}(n) - p_M(n)\,b_M(n) \tag{4.3.88}$$

Equations (4.3.85), (4.3.87) and (4.3.88) constitute the update equations for the joint estimation form of WLSL. We give all the update equations for WLSL algorithm in the joint process estimation form below.

(i) $$\Delta_{M-1}(n) = \lambda\,\Delta_{M-1}(n-1) + \alpha_n \frac{b_{M-1}(n-1)\,f_{M-1}(n)}{\gamma_{M-1}(n-1)}$$

(i)' $$\Delta_{M-1}(n) = \lambda\,\Delta_{M-1}(n-1) - \frac{b_{M-1}(n-1)\,f_{M-1}(n)}{\frac{1}{\alpha_n} - \frac{1}{\alpha_{n-1}} + \frac{\gamma_{M-1}(n-1)}{\alpha_{n-1}}}$$

(ii) $$\Gamma_{f,M}(n) = \frac{\Delta_{M-1}(n)}{B_{M-1}(n-1)}$$

(iii) $$\Gamma_{b,M}(n) = \frac{\Delta_{M-1}(n)}{F_{M-1}(n)}$$

(iv) $$f_M(n) = f_{M-1}(n) - \Gamma_{f,M}(n)\,b_{M-1}(n-1)$$

(v) $$b_M(n) = b_{M-1}(n-1) - \Gamma_{b,M}(n) f_{M-1}(n)$$

(vi) $$F_M(n) = F_{M-1}(n) - \frac{|\Delta_{M-1}(n)|^2}{B_{M-1}(n-1)}$$

(vii) $$B_M(n) = B_{M-1}(n-1) - \frac{|\Delta_{M-1}(n)|^2}{F_{M-1}(n)}$$

(viii) $$\gamma_M(n-1) = \gamma_{M-1}(n-1) - \alpha_n \frac{|b_{M-1}(n-1)|^2}{B_{M-1}(n-1)}$$

(viii)' $$\gamma_M(n) = \gamma_{M-1}(n) - \alpha_n \frac{|b_{M-1}(n)|^2}{B_{M-1}(n)}$$

(ix) $$\rho_M(n) = \lambda\, \rho_M(n-1) + \alpha_n \frac{b_M(n)\, e_{M-1}(n)}{\gamma_M(n)}$$

(x) $$p_M(n) = \frac{\rho_M(n)}{B_M(n)}$$

(xi) $$e_M(n) = e_{M-1}(n) - p_M(n)\, b_M(n)$$

Notice that in this set of update equations, (viii)' is always required because the quantity, $\gamma_M(n)$ is necessary in the joint estimation section in update equation (ix). Hence, if equations (i) and (viii) are used, one more variable and equation is required than if (i)′ and (viii)′ are used. In actual implementations, therefore, the primed equations are advantageous in terms of computational speed and memory requirements.

When the WLSL in the joint estimation form is used for adaptive noise cancellation, u(i) is the reference input and d(i) is the primary input. Remarks 1 through 4 made in connection with WLSL holds in the joint estimation case also. However, it should be noted that $\gamma_M(n)$ is a measure of how Gaussian u(i), the reference input, is and not d(i), the primary input. If primary input has outliers and other "bad" data, α_i will have to be pre-determined as mentioned earlier in Remark 4.

An important fact to be noted in the above derivation is that we have not explicitly made any stationarity assumptions. Neither has it been used implicitly like in using Toeplitz property of the correlation matrix or by equating forward and appropriately shifted backward prediction errors, etc.

In the next chapter, we shall explore the behaviour of WLSL in the joint estimation form by simulation and discuss its application to single stimulus BSAEP.

CHAPTER 5

SIMULATION & APPLICATIONS TO EVOKED POTENTIALS

5.1 Overview of the Chapter

Simulation studies were done to assess the performance of the weighted exact least squares lattice (WLSL) for adaptive noise cancellation. Although a large number of simulations were performed in the course of this research, only a few characteristic results which demonstrate the performance of WLSL and its superiority over normalized exact least squares lattice (NLSL) in adaptive noise cancellation applications are given in section 5.2. A performance index, M is defined and used in these studies. Results of earlier studies using NLSL are given in section 5.2.1. Here, we explain an unconventional choice for the forgetting factor, λ. In section 5.2.2, results for the weighted exact least squares lattice algorithm are given. The choice of α_i in the WLSL algorithm to obtain superior performance to that of NLSL is discussed and demonstrated by simulation. Finally, results of application of normalized LSL to real BSAEP data are given in section 5.3.

5.2 Simulation Studies

In this section, we present the simulation results of analysis of adaptive noise cancellation of AR processes using weighted and normalized exact least squares lattice algorithms. For both the normalized exact least squares lattice (NLSL) and weighted exact least squares lattice (WLSL) algorithms, we consider two cases where different weighting schemes are used. The objective is to demonstrate the differences in adaptive noise cancellation performance for these four conditions. To compare the performance of these algorithms for adaptive noise cancellation, we define a performance index, which we will call "M index".

M index is defined as the ratio of the mean square error between the desired signal and the primary input (which is variance of noise in primary input) to the mean square error between the desired signal and the estimated signal for any lattice order = M.

$$\text{M index} = 10 \log_{10} \frac{\text{mean square error}\big|_{\text{primary input}}}{\text{mean square error}\big|_{\text{order}=M}} \quad \text{decibels}$$

For lattice order = 0, the estimated signal is the result of scaled subtraction of the reference input from the primary input (note that scaling is not constant over all time indices, i, but varies adaptively to minimize the performance criterion, as explained in chapter 4). For proper lattice order which best fits the multiple linear regression model relating the noise in reference and primary inputs, the corresponding mean square error will be minimum and the M index will take its maximum value. Therefore, M index is a measure of performance of the lattice algorithm for order = M and we shall use this measure to compare various lattice algorithms for adaptive noise cancellation.

The next two sections give the details of simulation and the results. In section 5.2.1, we discuss some earlier work done (Madhavan et.al., 1984) using normalized exact least squares lattice algorithm (Lee et al, 1981). We consider two cases - (i) forgetting factor, λ = 1.00 and (ii) forgetting factor, λ = 1.03. In section 5.2.2, the performance of the lattice algorithm derived in this thesis, WLSL algorithm, is demonstrated. Two cases where (i) weighting factor, α_i is constant throughout and (ii) weighting factor, α_i is variable are presented. Comparisons are made among these four types - two for normalized and two for weighted exact least squares algorithms - and results presented at the end of section 5.2.2.

For simulation studies in sections 5.2.1 and 5.2.2, we used 256 samples of an AR(2) process, u(i).

$$u(i) = 0.4\, u(i-1) - 0.043\, u(i-2) + w(i)$$

where w(i) is white gaussian

This process was chosen for illustration only and similar results can be obtained for other time series. Referring to figure 4.1, u(i) is the reference input. To generate the primary input, d(i), the white gaussian sequence, w(i) is added to a signal at appropriate signal to noise ratios. The signal, s(i) is a zero-phase bandpass filtered (400 - 1500 Hz) brain stem auditory evoked potential. The multiple linear regression model relating noise in reference and primary inputs is of order 2 as shown by the above equation for u(i) and a second order lattice will be required for effective noise cancellation.

5.2.1 Normalized Exact Least Squares Lattice (NLSL)

Figure 5.1 shows all the data used in this simulation. Figure 5.1 a) shows 256 samples of reference input, u(i) which is a realization of the AR(2) process as given in the previous section. White gaussian noise, w(i) is added at 0 db signal to noise ratio to the signal shown in b) to result in c) which is the primary input. As mentioned before, the last 128 samples in b) is a zero-phase band pass filtered normal BSAEP. Referring to figure 4.8, the reference input, u(i) is shown in figure 5.1 a) and primary input, d(i) in figure 5.1 c). The lattice predictor and the transversal filter sections of the WLSL transforms d(i) based on the correlation between u(i) and noise in the primary input and the result is subtracted from the primary input. In adaptive noise cancellation, the estimation errors, $e_M(i)$ are the estimates of the signal in the primary input. In this section, the algorithm we have implemented is the normalized exact least squares lattice (NLSL) as

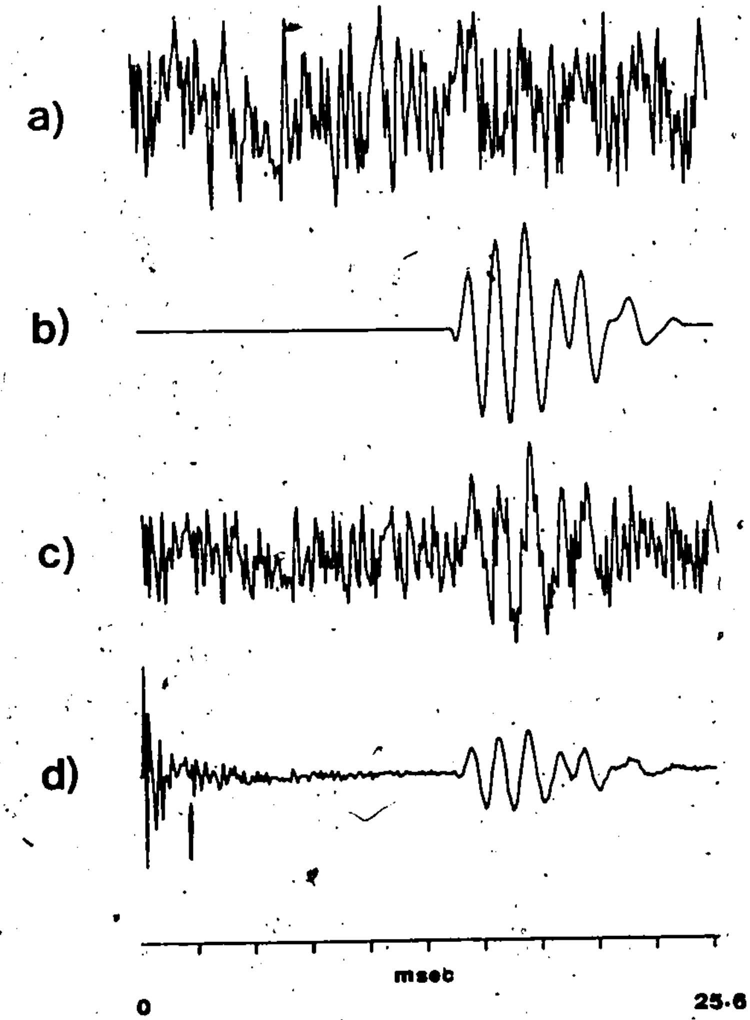

Figure 5.1 Simulation results for NLSL.

developed by Lee et al.(1981).

Figure 5.1 d) shows the estimated signal for lattice order 2 (or $e_2(i)$). This clearly shows adaptation at work and after about 100 samples, noise in the primary input is effectively removed. To investigate the effect of forgetting factor, λ, let us consider the role it normally plays in lattice algorithms. The value of λ is always chosen as less than or equal to 1.0. When $\lambda < 1.0$, for data that is non-stationary, the lattice algorithm 'forgets' the old values, in other words, weights remote past values much less than immediate past values giving an exponential weighting profile where the effective memory of the algorithm is approximately $1 / 1-\lambda$. In this simulation with $\lambda = 1.0$, the M index for 2nd order NLSL was 13.94 db. What we found was that by using $\lambda > 1.0$, the M index attainable for 2nd order NLSL is higher. For the estimated signal shown in figure 5.1 d), we used $\lambda = 1.03$ for the first 128 samples and $\lambda = 1.0$ for the last 128 samples. The M index for 2nd order NLSL was 20.4 db.

The use of $\lambda > 1.0$ can be explained as follows. Consider the signal in figure 5.1 b) as a 'transient disturbance' in the primary input. Knowing the position of occurance of this disturbance, we have chosen λ as stated in the previous paragraph. The resulting weighting profile is shown in figure 5.2. Because of the much larger weighting afforded to the first 128 samples compared to the last 128, the unwanted effect of transient disturbance in the adaptation procedure is minimized.

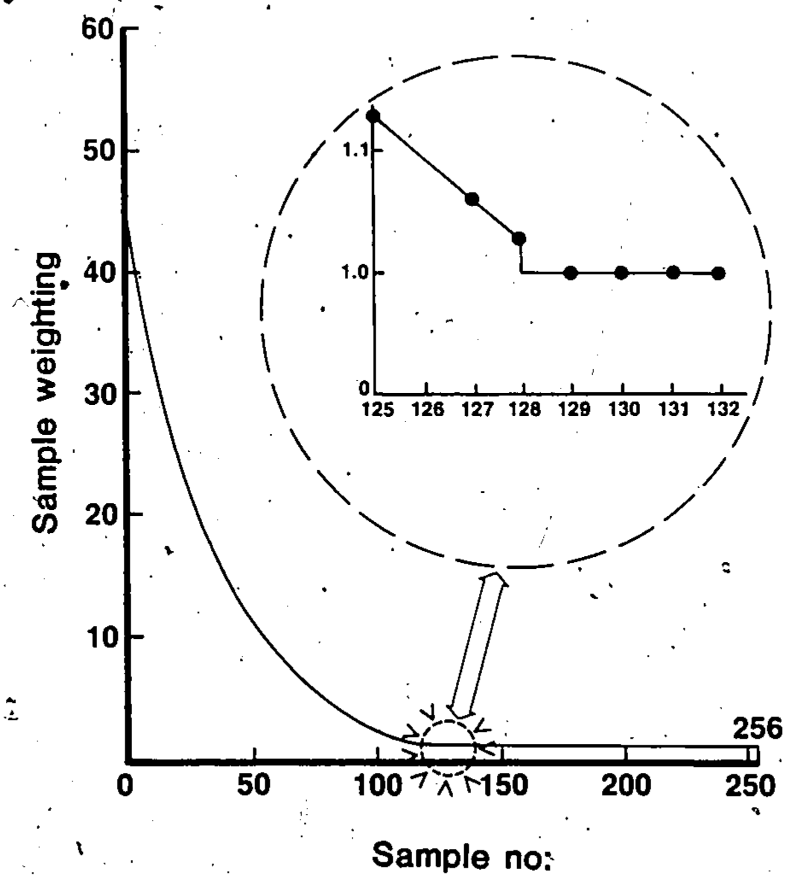

Figure 5.2 Weighting profile.

As mentioned in Chapter 4, the main motivation for developing a weighted exact least squares lattice algorithm stems from the desirability to weight the data independently of the forgetting factor, λ. One of the undesirable features of keeping $\lambda > 1.0$ is that the performance criterion that we are trying to minimize in the least squares problem, which are of the form -

$$E(n) = \sum_{i=1}^{n} \lambda^{n-i} \alpha_i \; |e(i)|^2$$

and other weighted energy terms can become unbounded as time index, n becomes larger and larger. In the present discussion of NLSL, however, that has been avoided by making sure that λ is greater than 1.0 only for a limited period of time. Given that different weighting is desirable, a weighted exact least squares lattice algorithm is necessary to accomplish this.

5.2.2 Weighted Exact Least Squares Lattice (WLSL)

Using the same AR(2) model and number of samples (256) as in the case of NLSL, detailed simulations were done with WLSL algorithm. Figure 5.3 shows a representative set of results from detailed simulations done. The complete results for WLSL are tabulated in table 5.1.

When weighting factor, $\alpha_i = 1.0$ for all i, the WLSL is the well-known unnormalized exact least squares lattice algorithm.(see remark 1. in section 4.3.4). The plots in figure 5.3 correspond to this choice of α_i. The top left-hand corner shows the AR(2) reference input. In the top right-hand corner is the signal that was added to white noise at different signal to noise ratios to give primary inputs. Figure 5.3 a), b) and c) correspond to primary input signal to noise ratios of +10 db, 0 db and -10 db, respectively. The left-hand column shows the primary inputs and the right-hand column, the estimated signals which are the estimation errors of 2nd order lattice.

It can be seen that the signal has been recovered very effectively. Indeed, there is very little indication of the initial adaptation other than the little kinks at beginning of the estimated signal plots. M indices for each of these cases were 12.66 db, 20.51 db and 24.31 db for signal to noise ratios, +10 db, 0 db and -10 db, respectively. As can be noticed in the left-hand column, for high signal to noise ratios, the signal appears as a 'transient disturbance' in the primary input. So as not to let this affect the lattice parameters, ~~in the~~ weighted exact least squares lattice algorithm, α_i can be properly chosen. Since most of the adaptation has taken place in the first 100 samples, the lattice can be shut down for the second half of the data set (a priori knowledge of when the signal is going to appear in the primary input is necessary). The lattice can be shut down (see remark 3. in section 4.3.4) conveniently by setting $\alpha_i = 0.0$ for $i = 129$ to 256 (for $i = 1$ to 128, $\alpha_i = 1.0$).

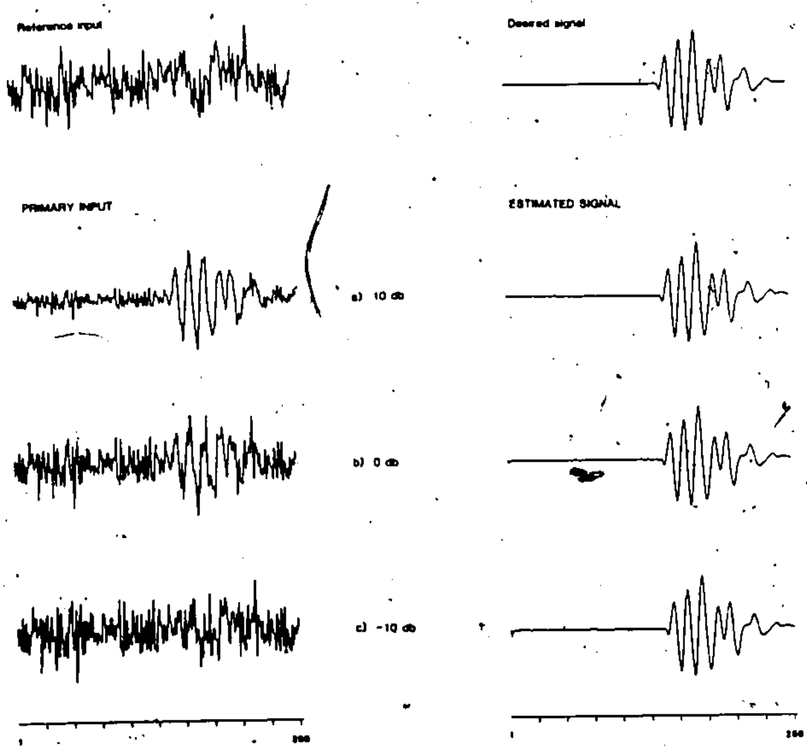

Figure 5.3 Simulation results for WLSL.

Using the weighting scheme stated above, detailed simulations were done for the same AR(2) process and 512 samples. Table 5.1 gives the results for +10 db to -60 db primary input signal to noise ratios. Comparing M indices for α_i = constant (= 1.0) to that for α_i = variable (lattice shut down for the second half), we can see that at high signal to noise ratios, elimination of the effect of 'transient disturbance' improves the performance of WLSL for adaptive noise cancellation dramatically. For example, at 10 db signal to noise ratio, M index when α_i is variable is 42.69 db where as when α_i is constant, M index is only 26.7 db. But, at low signal to noise ratios, the transient disturbance will be very small (very small signal variance in the primary input). Hence, the necessity to shut down the lattice is less. Besides, by shutting down the lattice, all feedback within the lattice is cut off and the performance of the WLSL is degraded. This can be seen for signal to noise ratios less than -20 db where M index for variable α_i is less than that for constant α_i. This trade-off has to be considered in practical applications.

An overall comparison is presented in figure 5.4. The same AR(2) process and 256 samples were used in comparing the performance of the following four cases of lattice alogorithms -

a) WLSL: α_i = variable (explained earlier)

b) WLSL: α_i = constant (= 1.0)

c) NLSL: λ = 1.03

d) NLSL: λ = 1.00

Table 5.1

Performance of WLSL algorithm. M index for constant and variable α_1 for different signal to noise ratios of primary input.

Signal to noise Ratio (db)	M index (db)	
	α_1 = constant	α_1 = variable
+10	26.7	42.69
+5	31.14	47.31
0	34.75	50.31
-5	37.16	50.02
-10	38.5	46.77
-15	39.19	42.26
-20	39.5	37.38
-25	39.59	32.41
-30	39.46	27.42
-40	38.02	17.52
-50	33.0	8.44
-60	24.49	2.77

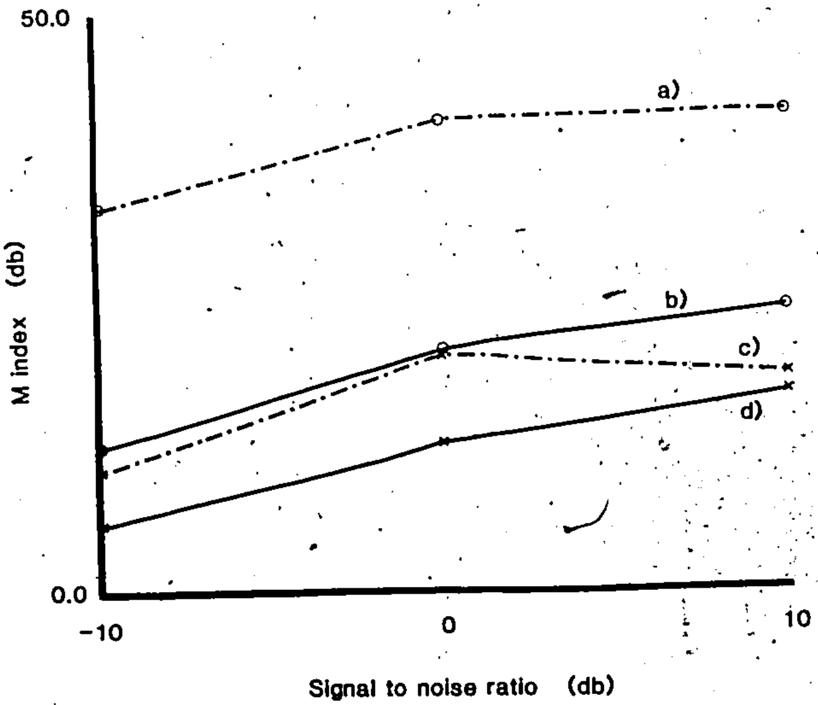

Figure 5.4 Performance comparison of WLSL to NLSL.

Use of $\lambda > 1.00$ gives better performance in the NLSL case, as can be seen in figure 5.4 c). The improvement for b) over d) has been mentioned in literature (Reddy et al., 1981) but no reason was given in that reference. However, they go on to conjecture that "normalized algorithms might prove to be advantageous in situations with less accuracy in computations". At this point, we do not have much to add to their comments.

The improvement in performance using variable α_i in the WLSL alogrithm is significant as seen in figure 5.4. The top trace a) represents the variable α_i case. Restating the conditions for the use of variable α_i, it is advantageous to make the lattice insensitive to unwanted or bad data (the signal in the primary input falls into this category since for adaptive noise cancellation, the primary function of the lattice is to learn the relationship between the noise in the reference and primry inputs) by reducing or even setting equal to 0.0 (corresponding to shutting down the lattice) the value of α_i. It can be concluded that weighting allows us to make the exact least squares lattice algorithm insensitive or robust to unwanted or bad data in the inputs of the lattice.

Remark 1. To implement the WLSL algorithm, a 'time exchange' computer code can be written. In this program, only the present and previous values of some of the internal parameters of the lattice need be stored. Pack and Satorious (1979) give such a Fortran program. By way of caution, one should mention that the programming of lattice algorithms is very tricky, in that even when some of the internal parameters of the lattice are not properly updated, the lattice will give apparently adequate results!

Remark 2. Number of computations required for lattice algorithms in general, is relatively high. Cioffi and Kailath (1984) gives a comparison for various algorithms. In particular, NLSL algorithm in the joint estimation form requires 29N operations (multiplies and divides) per iteration where N is the order of the lattice. The WLSL algorithm presented here requires 25N operations (11 divides) in the joint estimation form. With special purpose computational accelerators, this large number of computations can be performed in a reasonable length of time.

Remark 3. The number of data samples used will depend on the model of the reference and primary input processes. For example, when the poles of the model are close to the unit circle, larger number of samples than were used in this simulation study (256 and 512) will be required.

Remark 4. In our simulation studies, since the reference input was AR(2) and primary input was white, only a finite impulse response lattice was necessary. If the primary input was also

AR, an infinite impulse response lattice will be required (ratio of two polynomials). Methods to achieve this are discussed in Friedlander (1982 a). However, in practice, this infinite impulse response is approximated by a lattice of large order (Friedlander, 1982 a; Gardiner et al., 1985), usually 100 or more.

Remark 5. Ljung and Ljung (1985) have studied the numerical stability of lattice algorithms. They conclude that lattice algorithms are exponentially stable in that the effects of error decay exponentially. Other studies done by Samson and Reddy (1983) and Swanson and Symonds (1984) on the stability of normalized lattice for fixed point implementations have shown certain instances of instability. Unnormalized lattices are always numerically stable (Wiberg et al., 1985). Word length effects for floating point implementations have not been discussed in the literature. Although no detailed analysis has been done, our experiences with NLSL and WLSL suggest that the problems due to word length effects are more severe in the case of WLSL than NLSL. Word length has to be an important consideration in the implementation of the weighted exact least squares lattice algorithm.

5.3 Applications to Evoked Potential Estimation

The technique of adaptive noise cancellation using the normalized exact least squares lattice algorithm was applied for the estimation of brain stem auditory evoked potentials (BSAEP). Initially, a conventional averaged BSAEP was collected from the subject using Nicolet Pathfinder II equipment for later comparison with single stimulus (SSt) BSAEP. For the single stimulus procedure, data filtered between 150 and 2000 Hz were digitized at 10 KHz. The primary input to the lattice consisted of 12.8 msec of post-stimulus e.e.g. and 12.8 msec of e.e.g. immediately preceding the stimulus (figure 5.5). The reference input was 25.6 msec of e.e.g. immediately preceding the pre-stimulus e.e.g. in the primary input (figure 5.6). All the signals are succeeding segments of e.e.g. from the same electrode location, C_z.

Restating the assumptions for adaptive noise cancellation, the e.e.g. in the reference and primary input are assumed to be correlated and the evoked potential in the primary input immediately following the stimulus for 12.8 msec (or at least the first 7 msec where the evoked potential of interest occurs) and e.e.g. are assumed to be uncorrelated. Some justifications for the above assumptions were discussed in Chapter 2 (section 2.2 on the origin of evoked potentials). BSAEP, being a short latency EP, the e.e.g. in the first 10 msec is hardly desynchronized (effect of stimulus is not yet present in the e.e.g.). Hence, the assumption that BSAEP is uncorrelated to e.e.g. is tenable. The filtered e.e.g. used in this study has its peak power in the range 100 - 200 Hz. This would imply that e.e.g. in the reference and primary inputs which are separated by 25.6 msec are correlated.

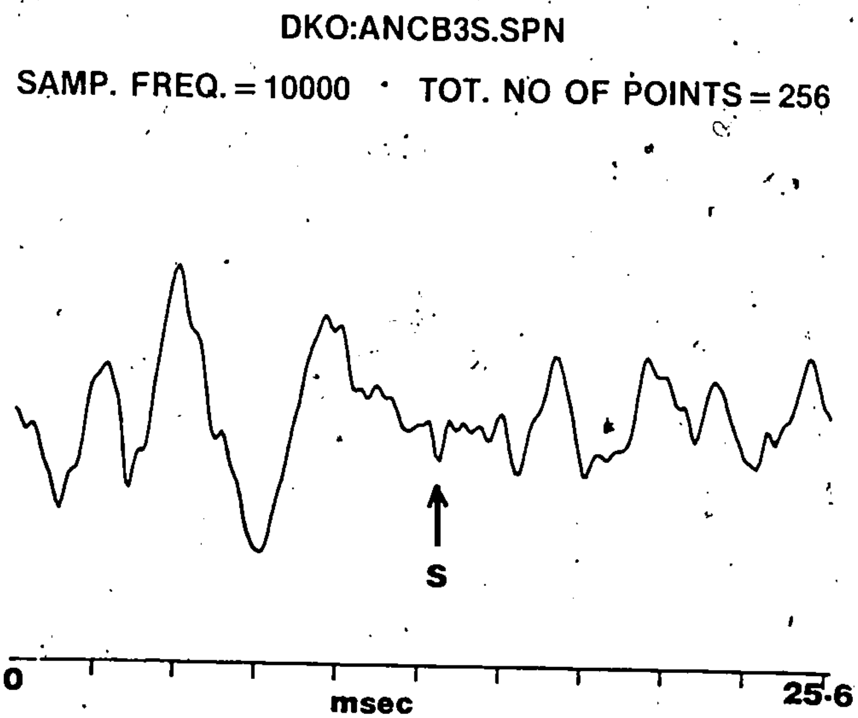

Figure 5.5 Primary input. 'S' - point at which stimulus was delivered.

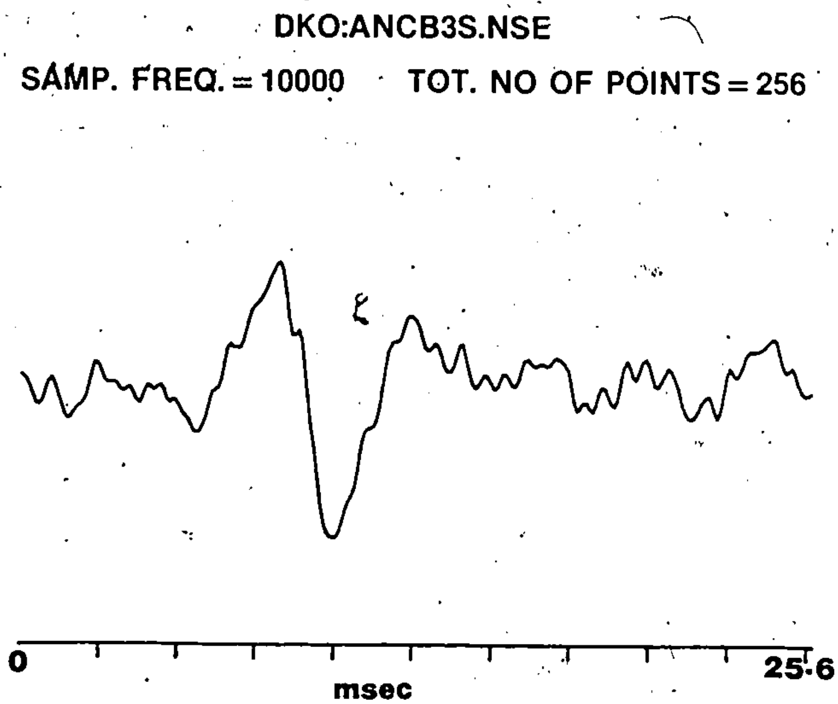

Figure 5.6 Reference input.

For adaptive noise cancellation using NLSL, forgetting factor λ, was chosen equal to 1.03 for samples 1 to 128 and 1.00 for samples from 129 to 256. The effect of this choice was discussed in section 5.2.1 and at the end of section 5.2.2. Since e.e.g. in the reference and primary inputs can be modeled as high order AR processes (order of 5 or 6 for the e.e.g. used in this study), it is required to have a high order lattice for noise cancellation, as discussed under remark 4. in section 5.2.2. The constraints of the computer system that we were using (PDP 11/34) allowed only upto 50th order lattices.

Figure 5.7 shows 100 samples (immediately following the stimulus) of the estimated signal for the 50th order lattice. Peaks I to V have been identified on this SSt BSAEP and notice the clear formation of the peaks except for the small bump between peaks III and IV. Figure 5.8 shows a comparison of conventional averaged BSAEP (ensemble average of 2000 responses) and SSt BSAEP. It can be seen that the peaks in the top trace (averaged BSAEP) and in the bottom trace (SSt BSAEP) do not match exactly. This is not particularly surprising since the top trace is the average of 2000 random realizations whereas the bottom trace is one such realization (note that in the entire procedure of adaptive noise cançellation, we do not make the assumption that the signal (BSAEP) is deterministic, ie., each of the 2000 stimuli produce identical evoked responses).

Eventhough, this is a very encouraging result, we hasten to add that extensive trials could not be done due to the limitations of the computer system we were using. Based on the simulations done, as is

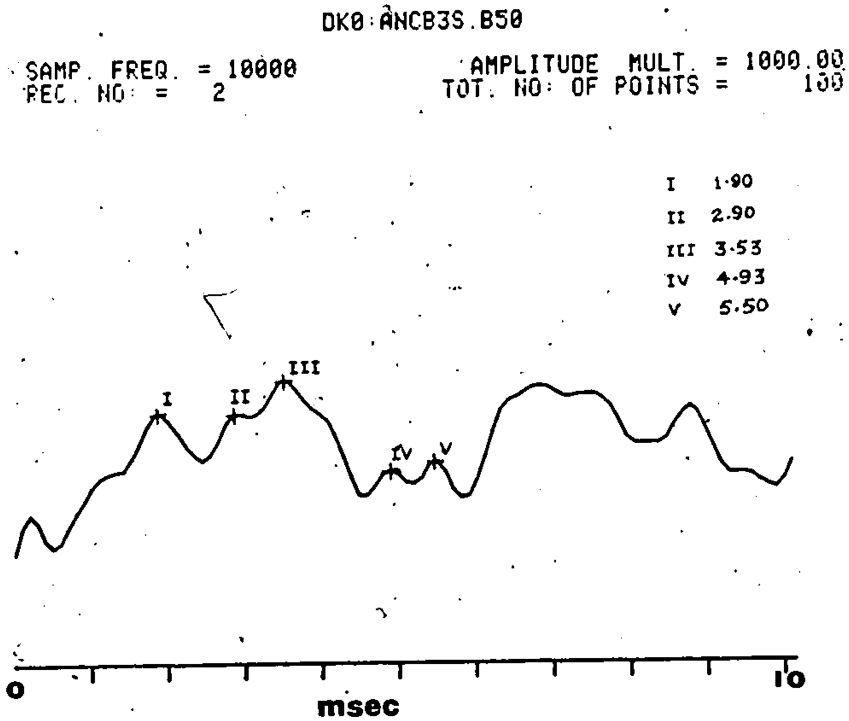

Figure 5.7 Single stimulus BSAEP.

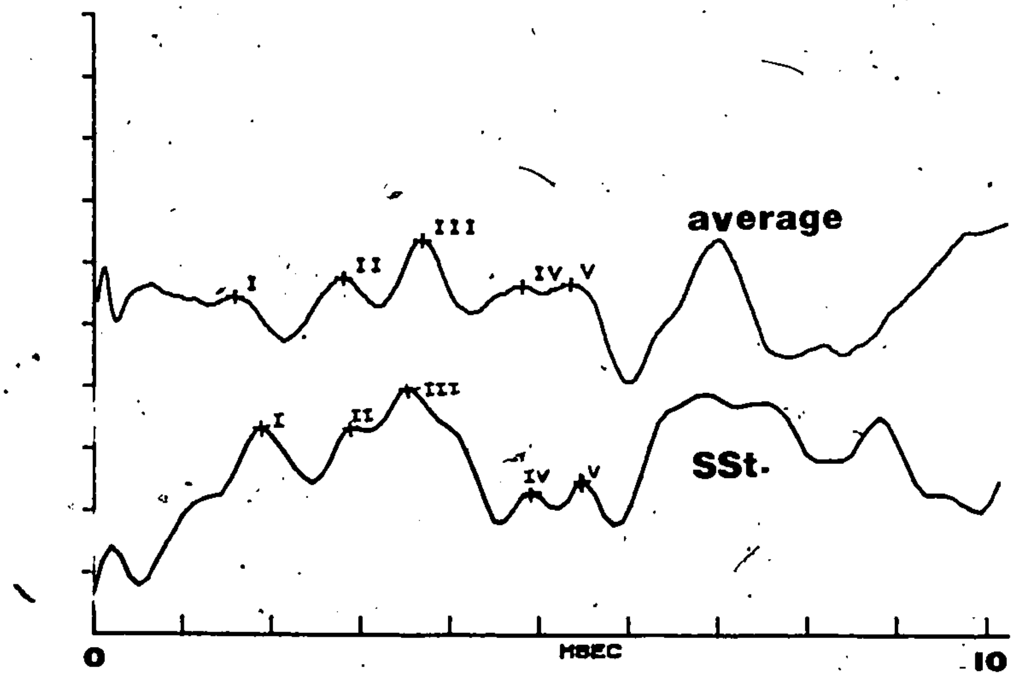

Figure 5.8 Comparison of ensemble averaged and single stimulus BSAEP.

evident from the results presented at the end of section 5.2.2, we feel that noise cancellation to extract SSt BSAEP using WLSL with variable weighting, α_i could be more effective. In the light of remarks 4. and 5. in section 5.2.2, we could not test this conjecture using our computer facilities which necessitated special program development and testing. Such development is underway at the present time. However, we feel that using WLSL with variable α_i will be preferable since, as seen in figure 5.4, its performance is significantly better than NLSL with λ = 1.03.

We would like to emphasize that this testing with real data was mainly designed to demonstrate one of the many possible paradigms for single stimulus EP extraction and the power of exact least squares lattice algorithms for adaptive noise cancellation. It is by no means an optimized solution to the problem of extraction of single stimulus evoked potentials. However, the simulation studies and real data test have shown that adaptive noise cancellation using exact least squares lattice alogrithm can be used to extract single stimulus evoked potential.

P A R T - C

PATTERN RECOGNITION

CHAPTER 6

SYNTACTIC PATTERN RECOGNITION OF EVOKED POTENTIALS

6.1 Overview of the Chapter

In view of the structural specificity of the BSAEP waveform, we employ syntactic methods for its pattern recognition and classification. Details of preprocessing by zero-phase bandpass filtering, primitive extraction, the finite-state grammar chosen and methods for parsing and classification are given in section 6.2. A training run on seventy patients was used to fine tune the system, to arrive at an empirically optimized classification criterion and to build a data base of normal BSAEP parameters. The method is discussed in section 6.3. In section 6.4, the results of a test run on sixty patients of unknown diagnosis are given. We conclude the chapter with a discussion of the results in section 6.5. Work presented in this chapter has been reported in the literature (Madhavan et al., 1986).

6.2 Theory of Syntactic Method

In present day neurological practice, the clinician identifies the peaks by visual inspection and classifies the BSAEP as normal or abnormal based on its appearance. The quatitative guidelines in this classification procedure are mainly the latencies of the peaks. Some researchers have established normal limits for these latencies (Kiloh et al., 1981; Fridman et al., 1982; Rose Mary Edwards et al., 1982). Selective attenuation of some peaks is also indicative of abnormality. Considering the structural specificity of the BSAEP, the procedure can be best automated by syntactic pattern recognition (Fu, 1982; Stockman et al., 1976). An idealized BSAEP is shown in figure 6.1.

The main motivation for automatic pattern recognition and classification of BSAEPs arises out of a need to supplement the clinician's visual analysis with a more objective method. A syntactic recognition method not only classifies the pattern but also builds up a description of the pattern. In the case of BSAEP, the description is in terms of peak locations. Such an objective criterion for peak selection allows the comparison of peak latencies and amplitudes between subjects and between centers. The unsupervised nature of the pattern recognition scheme makes it ideal for screening at remote locations, monitoring in intensive care situations and during surgical procedures.

The syntactic pattern recognition system consists of three steps as discussed by Fu (1982).

(i) Preprocessing where the pattern under consideration is filtered, restored and enhanced.

(ii) Primitives that have been decided upon which will adequately represent the signal, are extracted.

(iii) Syntax analysis involving the decision as to whether

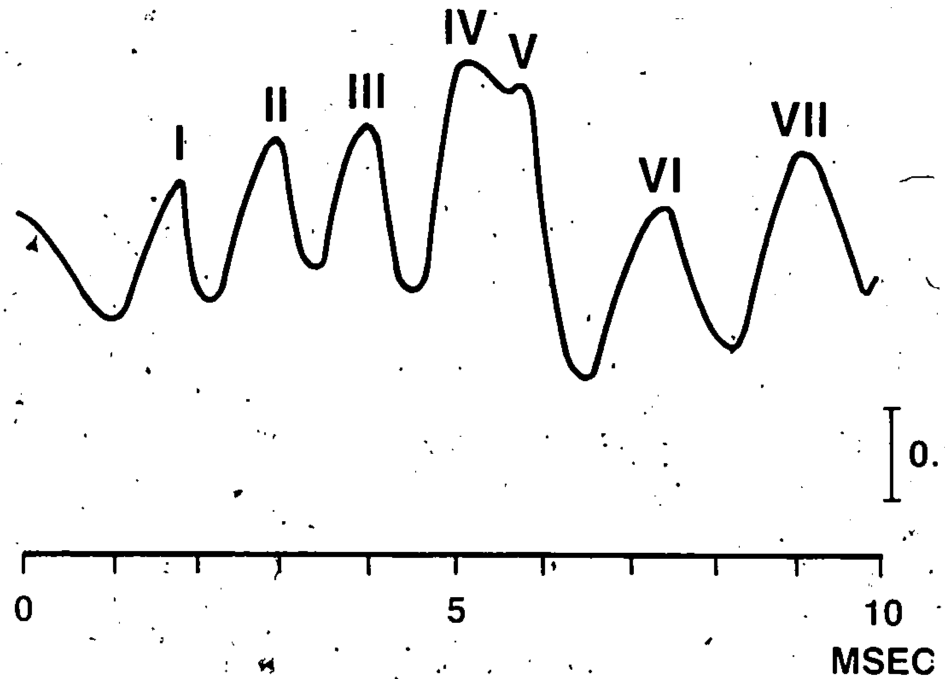

Figure 6.1 Idealized BSAEP.

or not the string of primitives is syntactically correct (ie., belongs to the class of patterns described by the given syntax or grammar).

We will look at each of these steps in turn in the context of BSAEP, which will clarify the procedure.

6.2.1 Preprocessing

BSAEPs used in this study were randomly selected from the large number of normal and abnormal responses obtained from routine clinical assessments. A PDP 11/34 minicomputer was used to digitize and process these responses. Five hundred data points corresponding to 10 msec (sampling rate of 50 Khz) were stored for each BSAEP.

Earlier studies (Fridman et al., 1982; Boston, 1981) have shown that BSAEP spectral components in the range 400 to 1500 Hz determine the location of its peaks. As a means of smoothing the data, a zero-phase bandpass filter in the above range was designed using the method of McClellan et al. (1973). Zero-phase filtering is extremely important in this context because of the significance of the latencies of the peaks (Boston and Ainslie, 1980; Doyle and Hyde, 1981). A filter with a pass band of 400 to 1500 hz and 97 coefficients was designed. BSAEP is convolved with the filter coefficients to give the preprocessed signal. Peaks I to V being most important, data points 50 to 350, corresponding to 1 to 7 milliseconds, were considered the active region and set up for further analysis.

6.2.2 Selection of Primitives

The choice of primitives, i.e., basic signal descriptors, to repesent the pattern is the key to success of the syntactic method. Primitives must be general enough so that the pattern can be represented without too many of them, yet they should be capable of preserving all the required shape information of the BSAEP. We selected the following three primitives.

$$\nearrow \quad - \quad \text{'a'}$$

$$\searrow \quad - \quad \text{'b'}$$

$$\rightarrow \quad - \quad \text{'c'}$$

where arrows indicate the direction of the signal slope.

Only general directions rather than exact gradients are considered in the extraction of primitives. Each signal sample, y(i) can be replaced by a primitive according to the following rules:

$$y(i+1) - y(i) \quad > \quad \varepsilon \quad \longrightarrow \quad y(i) \text{ replaced by 'a'}$$

$$y(i+1) - y(i) \quad < \quad \varepsilon \quad \longrightarrow \quad y(i) \text{ replaced by 'b'}$$

$$|y(i) - y(i+1)| \quad \leq \quad \varepsilon \quad \longrightarrow \quad y(i) \text{ replaced by 'c'}$$

The bound 'ε' is chosen as a selectable percentage of the maximum difference between any y(i) and y(i+1). The beginning of the sentence or string of primitives is defined as the point after which there is a selectable number (IBEG) of consecutive 'a's indicating the start of the first peak of the EP.

6.2.3 Grammar

Investigations into the mathematical structure of natural languages, aimed at trying to understand its properties, revealed that primitives (alphabet) with a set of rewriting rules, the combination of which is called a grammar, can be used as a method of describing languages (Fu, 1982). In our context, a grammar should describe one class of patterns (and one alone), such as normal BSAEPs, when primitives have been defined as in the previous section. A grammar G, is defined as a set of four entities - a starting symbol S, non-terminals V_n which are some combinations of primitives, terminals V_t which are primitives and a set of production rules P which relate S to V_t through V_n.

BSAEP can be considered as a succession of five 'HILLS' with a HILL defined as in figure 6.2. Sharp and flat hills are acceptable. A finite-state or type 3 (Fu, 1982) grammar G_{Hill}, described below, characterizes a hill.

G_{Hill}:

S - the starting symbol, 'HILL'
V_n - the non-terminals, [A,B,C] which are the intermediate steps in the relationship between S and V_t
V_t - the set of primitives, [a,b,c] as defined previously
P: - the set of production rules which, when applied, generates sentences (strings of primitives) from starting symbol, 'HILL'

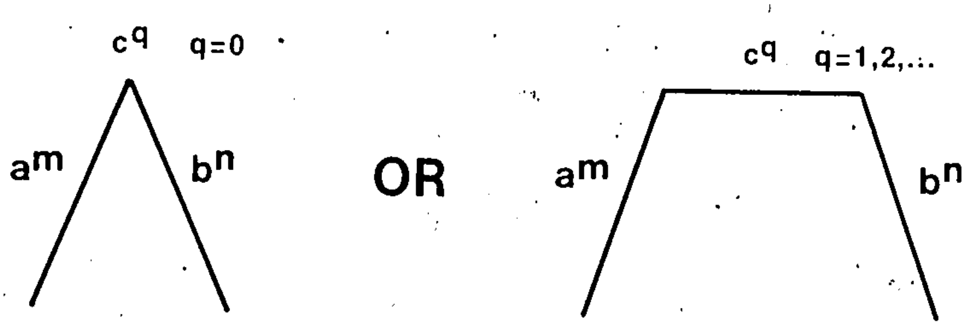

Figure 6.2 Definition of a 'HILL'.

(Arrows below denote "can be replaced by")

1.	HILL	$\longrightarrow$	aA
2.	A	$\longrightarrow$	aA
3.	A	$\longrightarrow$	cB
4.	A	$\longrightarrow$	bC
5.	A	$\longrightarrow$	b
6.	B	$\longrightarrow$	cB
7.	B	$\longrightarrow$	b
8.	B	$\longrightarrow$	bC
9.	C	$\longrightarrow$	bC
10.	C	$\longrightarrow$	b

Let us consider the grammar G_{Hill} and see how it can be applied to our recognition problem.

Starting with 'HILL',

(1) 'HILL' can be replaced by aA (rule 1.) giving $\longrightarrow$ aA

(2) A in aA can be replaced by cB (rule 3.) giving $\longrightarrow$ acB

(3) B in acB can be replaced by b (rule 7.) giving $\longrightarrow$ acb

This is an acceptable shape as shown in figure 6.2 since it contains in sequence, an upslope ('a'), a flat top ('c') and a downslope ('b'). It can be shown by enumeration that application of the production rules in any order after rule 1. has been applied, will generate only sentences of the form $a^m c^q b^n$ ($m,n \geq 1$, $q \geq 0$). This grammar can be iterated five times to identify the five hills in the BSAEP.

6.2.4 Parsing

Parsing is the step where we ascertain that the unknown BSAEP, after being preprocessed, is described by the grammar that we have chosen above. To facilitate parsing, we set up a table as shown below, filling up the cells with the number of occurences of the primitives for each peak. Rows are the primitives a, c, b and c in that order and columns are the I to V peaks. Each entry in the table, n_{ij} is the number of ith primitives in the jth peak.

PEAK

PRIMITIVE		1	2	3	4	5
'a'	1					
'c'	2			n_{ij}		
'b'	3					
'c'	4					

Parsing involves checking the entries in this table. The first and third rows indicate the positive and negative slopes of the hill. They must be non-zero to have the shape of a hill. The second row indicates the flatness of the hill.

6.2.5 Attribute Checking

Once the hills are recognized, their attributes have to be checked to see if they are acceptable (normal or abnormal). Latency of any peak p is

$$L_p = \frac{1}{50}\left[IBL + \left(\sum_{j=1}^{p-1}\sum_{i=1}^{4} n_{ij}\right) + n_{1p} + \frac{n_{2p}}{2}\right] \text{msec}$$

where IBL is the number of data points in the sequence before the beginning of the sentence. Using this latency, the peak amplitude can be determined from the original data set. The method to determine whether these attributes are acceptable is described in the next section.

6.3 Training Run

BSAEPs were collected in the neurological clinic of the McMaster University Medical Center using Nicolet Pathfinder II evoked potential equipment. The subjects used in the study (training and test run combined) had a mean age of 37.5 years with a standard deviation of 14.8 years. The routine clinical procedure of BSAEP collection was used. Biphasic (rarefaction and condensation) clicks were delivered to the ear of the subject at 9.9 repetitions per second. Clicks were of 100 microsecond duration and around 65 db above hearing threshold. The e.e.g. from C_z - A_2 montage with A_1 as reference was suitably amplified and bandpass filtered (150 - 3500 hz). After artifact rejection (based on amplitude criteria, which is a standard feature of the equipment), two thousand responses were averaged to obtain each BSAEP.

Preprocessing and pattern recognition computer programs were written in Fortran IV and implemented on a PDP 11/34 computer. A few initial runs were made to fine tune the parameters of the pattern recognition procedure. The bound 'ε' in the primitive extraction step was chosen as 3% of the maximum difference between any y(i) and y(i+1). It was found that a choice of IBEG = 10 was suitable (IBEG is the number of consecutive 'a' primitives necessary to define the beginning of the sentence). Primitives are extracted from the preprocessed BSAEP and the parse table constructed. If parse is succesful, ie., all five peaks have been identified, peak latencies are computed using the equation above and displayed along with the unfiltered BSAEP (figure 6.3). If they are within acceptable limits (procedure for their selection is described in the next section), the amplitudes of the peaks are found from the unfiltered data. When there is a neurological abnormality in the brain stem region, only that peak which corresponds to the area of abnormality is attenuated (Kiloh et al., 1981). Hence if all peaks fall within 25% of the average peak amplitude, the amplitude attribute check is successful. If the latency and amplitude attribute checks are successful, the BSAEP is labelled 'parse successful: attributes normal'.

Figure 6.4 shows unfiltered and filtered BSAEP and the final output of the pattern recognition program for a normal subject. The parse table and the sentence that was parsed are shown in figure 6.5. As described in the parsing section, rows 1 and 3 have non-zero elements indicating that there was a positive and negative slope giving the shape of a hill. All the five hills are flat-topped with one horizontal primitive 'c' in the 2nd row. From each of the columns in the table corresponding to each hill, the sentence can be written as shown at the

EP SUBJECT NO: 1H

PEAK 1 LATENCY = 1.560
PEAK 2 LATENCY = 2.540
PEAK 3 LATENCY = 3.620
PEAK 4 LATENCY = 4.700
PEAK 5 LATENCY = 5.520

* PARSE SUCCESSFUL : ATTRIBUTES NORMAL *

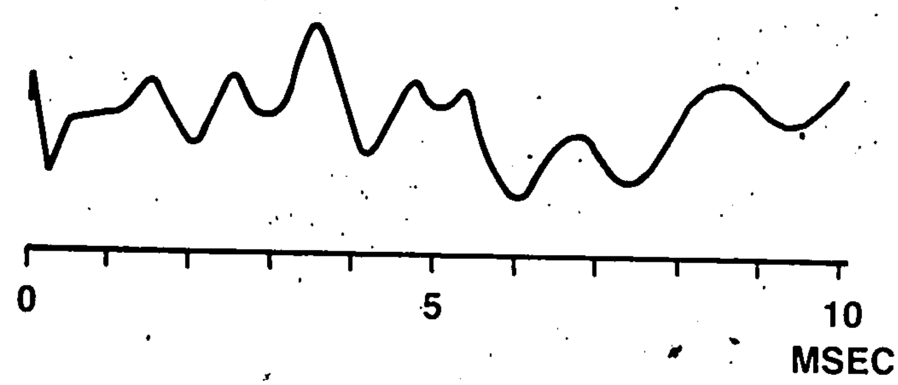

Figure 6.3 Final output of the pattern recognition procedure.

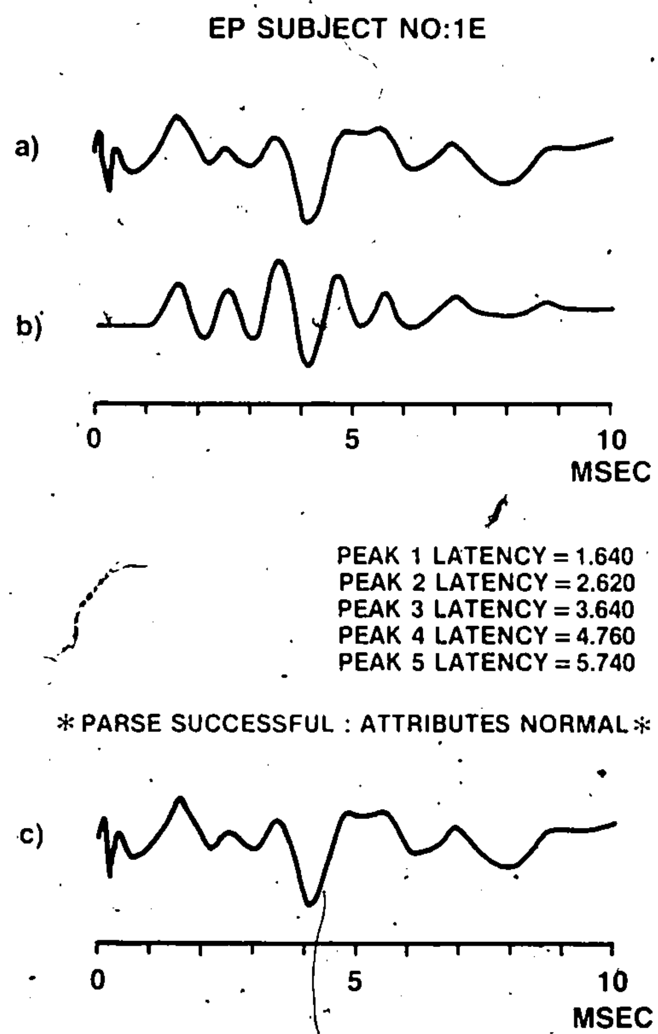

Figure 6.4 Zero-phase bandpass filtered BSAEP along with the final output of the pattern recognition procedure.

PARSE TABLE

31 1 25 1	22 1 23 1	26 1 27 1	27 1 25 1	22 1 24 2
Hill I a^{31} c b^{25} c	Hill II a^{22} c b^{23} c	Hill III a^{26} c b^{27} c	Hill IV a^{27} c b^{25} c	Hill V a^{22} c b^{24} c^2

Figure 6.5 Parse table and the sentence that was parsed.

bottom of the figure. For example, in the case of hill III, it has 26 'a' primitives (1st row of 3rd column), one 'c' primitive (2nd row of 3rd column), 27 'b' primitives (3rd row of 3rd column) and one 'c' primitive (4th row of 3rd column).

If, however, the parse fails, that message is displayed and peak latencies and amplitudes are not computed. If the parse is successful but either attribute check fails, peak latencies and the 'attribute mismatch' message is displayed. If either the parse fails or there is an attribute mismatch, the classifier considers that BSAEP 'abnormal'.

Fifty eight normal and 12 abnormal BSAEPs based on neurological assessment, were chosen randomly from our clinical records. In our previous study (Madhavan et al., 1983), the latency attribute check was based on mean absolute peak latencies and two standard deviations was selected as constituting an acceptable range for each peak latency. One problem with this approach is that a delay in one peak affects all the subsequent peak latencies. To avoid this undesirable carry-over effect, we used mean peak latency differences between successive peaks which then lets us treat them as 'uncorrelated'. Further, the latency of the first peak is not used in the classification at all because it is affected by extraneous factors, eg., wax in the ear. A database was created for mean peak latency differences and standard deviations of normal BSAEPs.

To determine what multiple of the standard deviation about each mean latency difference constitutes our acceptable range of normal values (rather than a fixed value of two as was chosen for our previous study (Madhavan, et al., 1983)), an empirical optimization run was performed. The multiplier was varied independently for each mean peak latency difference from 0.1 corresponding to a very tight range of acceptable values about the mean to 3.0 corresponding to a large range. Then the classifier performance was compared to the neurological assessment.

The following scoring system was used:

1. Concurrence of neurological and classifier results: +1
   (normal or abnormal)

2. Neurological normal; classifier abnormal : -1
   (false positive)

3. Neurological abnormal; classifier normal : -10
   (false negative)

A false negative was penalized more to reduce the danger of the classifier falsely identifying an abnormal response as normal and thus possibly escaping a closer scrutiny by the neurologist. It must be noted that the somewhat arbitrary scoring system that we have used can be modified according to the situation.

For each peak, the standard deviation multiplier that maximized the total score for all the 70 subjects was selected for use in the test run. The final values are as follows.

Peak Latency Difference	(II-I)	(III-II)	(IV-III)	(V-IV)
Multiplier	1.74	1.90	2.33	1.77

6.4 Test Run

Sixty BSAEPs of unknown (to the authors) clinical assessment were processed in this run. The procedure was exactly the same as before except that the optimized standard deviation multipliers were used. If the BSAEP was classified by the neurologist as normal, the database was updated. The final results for 60 test subjects which contained 35 normals and 25 abnormals are shown in table 6.1.

With a typical tertiary clinic prevalence of 42% (25 abnormals out of 60), we see that the accuracy of classification was 83%. The sensitivity, ie., the classifier's ability to detect abnormality when present, was 84% or 21 out of 25 times. The classifier correctly identified the absence of abnormalities 28 out of 35 times or the classifier's specificity was 80%. The total of 93 entries in the database (58 normals from training and 35 normals from test runs) yielded the mean peak latency differences and their standard deviations shown in table 6.2.

6.5 Discussion

These results show that the syntactic pattern recognition approach produces very acceptable results. Preprocessing by the zero-

Table 6.1

Result of classification of 60 subjects.
(21+28)=49 correctly classified.
4 false negatives and 7 false positives.

Clinical Assessment

Classification by Computer		Abnormal	Normal
	Abnormal	21	7
	Normal	4	28

Table 6.2

Statistics after 93 entries have been made into the data base of normal BSAEPs.
Mean values and standard deviations of relative peak latencies shown.
I-stimulus latency values not used in the classification procedure.

	Latency in milliseconds	
Peak Difference	Mean	Standard Deviation
I - stimulus	1.653	0.198
II - I	1.015	0.090
III - II	1.018	0.110
IV - III	1.075	0.080
V - IV	0.898	0.118

phase bandpass filter renders the rest of the steps in the pattern recognition procedure very easy. As shown in figure 6.4(b), the zero-phase filtered EP "looks" different from the classical EP as in figure 6.1 but because of the zero-phase nature of the filter, there is no distortion in the location of the peaks. Indeed, this is the feature used by the clinician in making the assessment and that is preserved by zero-phase filtering. A similar approach used earlier (Fridman et al., 1982) has also shown the efficacy of zero-phase bandpass filtering. In our application, the removal of noise and artifacts by preprocessing has made the grammar required to identify HILLS particularly simple. The finite-state grammar we have used here makes the parsing procedure easy. Software implementation, which chiefly involves logical operations, on any microcomputer will be straight-forward, making it inexpensive and portable.

One important feature of this classifier is the use of relative peak latency differences rather than absolute peak latencies from stimulus. By eliminating (at least partially) the carryover effect of delay in one peak affecting the latencies of subsequent peaks, we have achieved an improvement in classification accuracy from 75% in our previous study (Madhavan et al., 1983) to 83% in this case.

Instead of choosing a fixed acceptance criterion (say, 2 standard deviations about each peak latency), the multipliers were empirically optimized. In this procedure, a weighting system was developed which penalizes a false negative ten times more than a false positive. These weightings could be modified according to the application at hand. For example, in a screening situation where EP is only one part of a test battery, false positives (classifying normals as abnormals) could be penalized more. On the other hand, false negatives may be penalized much more than was done here ("-10") in an intensive care application when it is important to identify all abnormalities. It should be pointed out that as more and more entries are made in the database and the standard deviation values converge, the magnitude of the multipliers may have to be increased to avoid an increase in error rate of classification.

A sufficiently large sample of BSAEPs (seventy) was used to train the classifier. The test run had a typical tertiary care prevalence of abnormal BSAEPs and can be considered a valid test of the classifier. Accuracy of 83% ensures that the classifier performs adequately in a clinical context. Further improvement in accuracy of classification may be achieved by developing separate databases for infants, children, adults and geriatric groups as there are well-known differences in acceptable latencies for each of these groups. It is easy to modify the classifier for any other evoked potential such as visual and somatosensory potentials. Since it is general in nature, only the relevant attribute table need be established.

Our classifier is seen as a supplement to the clinician's visual analysis. Because of the objective method of locating peaks and classifying the EP, this algorithm will be useful where continuous attention of a trained neurologist is not available or possible as in EP screening in remote locations or in surgical and intensive care monitoring.

CHAPTER 7

CONCLUSIONS & RECOMMENDATIONS

In this thesis, we have presented some new and powerful methods for the estimation and pattern recognition and classification of evoked potentials. The ensemble averaging technique to estimate evoked potentials suffers from many drawbacks. The assumption that responses to each stimulus are identical, and hence the evoked responses are deterministic, is unrealistic. Since the transient responses are lost by averaging and there is an inherent time lag before all the responses in the ensemble are collected and averaged, we explored other methods for the estimation of the evoked potential.

Adaptive noise cancellation seemed to provide the best solution for the estimation of single stimulus evoked potentials. As many studies have shown (discussed in chapter 4), the popular LMS algorithm which has been used for noise cancellation has many drawbacks. In this thesis, we have demonstrated the use of a new family of adaptive algorithms, the so called lattice algorithms. Simulation studies and real data tests demonstrate the ability of the normalized exact least squares lattice algorithm as deveploped by Lee et al. (1981) to estimate single stimulus evoked potential.

These studies prompted us to develop a new and more general lattice algorithm called weighted exact least squares lattice algorithm. The complete theoretical derivation and simulation studies to demonstrate its superiority over normalized exact least squares lattice in adaptive noise cancellation applications is a major contribution of this thesis. The derivation of the algorithm, which parallels the procedure adopted by Haykin (1986) for unnormalized (unweighted) exact least squares lattice, shows the dissociation of gain vector, '$\underline{k}$' and 'γ' for forward and backward prediction. We explain the interpretation of 'γ' as a measure of how gaussian the input process is and its relationships to similar variables in published literature. The weighting factor, α_i in the weighted exact least squares lattice algorithm modifies the 'γ' appropriately to make the weighted exact least squares lattice algorithm robust. It is this property which gives it superiority over normalized exact least squares lattice alogrithm in adaptive noise cancellation applications.

We have assumed a very different approach to the pattern recognition and classification problem. Rather than extending the statistical approach taken in Part - B (adaptive filtering) of the thesis and using statistical pattern recognition techniques, we adopted syntactic methods which have their roots in Artificial Intelligence and Computer Science. This was dictated purely by the waveform that we were trying to classify. Evoked potentials being structurally specific waveforms and abnormality in them being very small variations in latency (corresponding to very small shifts in frequency and other parameters of the process), we felt that syntactic methods would be more appropriate for pattern recognition than statistical methods. The simplicity of the procedure and the performance of the classifier has demonstrated the validity of this premise. Another reason for the use of syntactic method was its ability to build up a description of the evoked potential in terms of its amplitude and latency which is not easily done with statistical methods.

The application of syntactic pattern recognition to evoked potentials has been demonstrated for the first time in this thesis. The finite state grammar we have developed to describe a waveform is unique, to this author's knowledge. The flexibility of this grammar is shown by the ease of modifying it to suit other applications (Nohara, 1985). The scoring system developed to optimize the classifier during the training phase can be tailored to other clinical situations by modifying them suitably. As a part of the classification procedure, a data base of normal evoked potentials is generated. The syntactic pattern recognition system was demonstrated to be practical by the exhaustive test run that was undertaken.

There are several directions in which the results developed in this thesis could be extended. To make the least squares lattice algorithm robust, the elegant methods developed by Puthenpura et al. (1986) can be incorporated into the weighted exact least squares lattice algorithm. Adaptive noise cancellation using other powerful adaptive algorithms such as Fast Transversal Filter (Cioffi and Kailath, 1984) and systolic arrays (Haykin, 1986) should be explored. Robustizing these algorithms will be a very useful extension to existing adaptive techniques. The use of joint estimation form of lattice algorithms for pattern recognition should be explored, where if the reference and primary inputs are feature vectors, the estimation error is a measure of the distance between them. This can form the basis of classification using distance measures. Another interesting extension of lattice algorithms is to the non-linear case. Lee (1980) mentions the possible extensions to Toda lattice which arises in the context of solitons (Toda, 1970). The interesting property of Toda lattices is their exponential nearest-neighbour coupling. This arises naturally in many physiological situations, especially in action potential propagation through axons and across synapses.

There is much more work to be done in the practical implementation of single stimulus evoked potential systems using adaptive noise cancellation. This work is currently underway at McMaster University. The practical solution has to be optimized in terms of filter order, number of signal samples, word length and computational power required. The question of alternate reference input sources is also not settled. Winsky and Allinson (1984) point out that coherence of alternate e.e.g. channels is low but we feel that other solutions can be found to this problem. The reference input to be used for other forms of evoked potentials such as visual, somatosensory and event related potentials has to be explored in detail.

The use of syntactic pattern recognition and classification can be readily extended to other evoked potentials. The generality of the grammar and the ease of implementation should make this extension readily possible. Once the technique is extended to other evoked potentials, they can be combined into a multi-modality evoked potential system which can form part of a neurological expert system. Different scoring systems should be tried to optimize the application of the classifier in situations such as screening in remote locations and monitoring in neurosurgical and intensive care situations. Partitioning of the data base of normal evoked potentials used in the classification procedure into different age groups should improve the accuracy of classification.

The results presented in this thesis have applications not only in neurological signal processing and biomedical engineering but also in adaptive filtering and pattern recognition and their applications in many areas of signal processing. We believe that our work will have a major impact in those areas.

REFERENCES

Ahmed, N. and Rao, K.R. 1975. Orthogonal transforms for digital signal processing, New York, Springer-Verlag.

Auerbach, V.H. and Haber, F. 1974. Two novel ways of averaging waveforms by fourier analysis: the representative and the synchronous averages. J. Franklin Inst., Vol. 297, 169.

Aunon, J.I. and McGillem, C.D. 1975. Techniques for processing single evoked potentials. Trans. San Diego Biomed. Symp., pp. 211-229.

Aunon, J.I., McGillem, C.D. and Childers, D.G. 1981. Signal processing in evoked potential research: averaging and modeling. CRC critical reviews in bioengineering, Bourne, J.R. (ed.), Florida, CRC Press.

Aunon, J.I. and McGillem, C.D. 1982. On the classification of single evoked potentials using a quadratic classifier. Computer Programs in Biomedicine, Vol. 14, pp. 29-40.

Aunon, J.I., McGillem, C.D. and O'Donnell, R.D. 1982. Comparison of linear and quadratic classification of event-related potentials on the basis of their exogenous and endogenous components. Psychophysiology, Vol. 19, No. 5, pp. 531-537.

Ayala, I.L. 1982. On a new adaptive lattice algorithm for recursive filters. IEEE Trans. ASSP, Vol. ASSP-30, No. 2, pp.316-319.

Basar, E. 1980. EEG-Brain dynamics, New York, Elsevier.

Berger, Hans. 1929. On the electroencephalogram of man. Psychophysiology, Porges, S.W. and Coles M.G.H. (eds.), 1976, Pennsylvania, Dowden, Hutchinson and Ross Inc., pp. 9-14.

Borda, R.P. and Frost, J.D. 1968. Error reduction in small sample averaging through the use of the median rather than the mean. Electroenceph. Clin. Neurophysiol., Vol. 25, 391.

Boston, J.R. and Ainslie, P.J. 1980. Effects of analog and digital filtering of brain stem auditory evoked potentials. Electroenceph. Clin. Neurophysiol., Vol. 48, pp. 361-364.

Boston, J.R. 1981. Spectra of auditory brainstem responses and spontaneous EEG. IEEE Trans. BME, Vol. BME-28, No. 24, pp.334-341.

Bourne, J.R., Jagannathan, V., Giese, B. and Ward, J.W. 1980. A software system for syntactic analysis of the EEG. Computer Programs in Biomedicine, Vol. 11, pp. 190-200.

Bourne, J.R., Jagannathan, V., Hamel, B., Jansen, B.H., Ward, J.W., Hughes, J.R. and Erwin, C.W. 1981. Evaluation of a syntactic pattern recognition approach to quantitative electroencephalographic analysis. Electroenceph. Clin. Neurophysiol., Vol. 52, pp. 57-64.

Burg, J.P. 1968. A new analysis technique for time series data. Modern Spectrum Analysis, Childers, D.G. (ed.), New York, IEEE Press, pp. 42-48.

Chiappa, K.H. 1983. Evoked potentials in clinical medicine. New York, Raven Press.

Childers, D.G. 1977. Evoked responses: electrogenesis, models, methodology, and wavefront reconstruction and tracking analysis. Proc. IEEE, Vol. 65, No. 5, pp. 611-625.

Childers, D.G., Bloom, P.A., Arroyo, A.A., Roucos, S.E., Fischler, I.S., Achariyapaopan, T. and Perry, N.W. 1982. Classification of cortical responses using features from single EEG records. IEEE Trans. BME, Vol. BME-29, No. 6, pp. 423-438.

Cioffi, J.M. and Kailath, T. 1984. Fast, recursive-least-squares transversal filters for adaptive filtering. IEEE Trans. ASSP, Vol. ASSP-32, No. 2, pp. 304-337.

Cohen, B.A. and Sances, A. 1977. Stationarity of the human electroencephalogram. Med. & Biol. Eng. & Comput., Vol. 15, pp. 513-518.

Dawson, G.D. 1947. Cerebral responses to elctrical stimulation of peripheral nerve in man . Neurol. Neurosurg. Psychiatr.. 10: 134.

Doyle, D.J. 1975. Some comments on the use of Wiener filtering for the estimation of evoked potentials. Electroenceph. Clin. Neurophysiol., Vol. 38, pp. 533-534.

Doyle, D.J. and Hyde, M.L. 1981. Bessel filtering of brain stem auditory evoked potentials. Electroenceph. Clin. Neurophysiol., Vol. 51, pp. 446-448.

Edwards, Rose Mary, Buchwald, J.S., Tanguay, P.E. and Schwafel, J.A. 1982. Source of variability in auditory brain stem evoked potential measures over time. Electroenceph. Clin. Neurophysiol., Vol. 53, pp. 125-132.

Fridman, J., John. E.R., Bergelson, M., Kaiser, J.B. and Baird, H.W. 1982. Application of digital filtering and automatic peak detection to brain stem auditory evoked potential. Electroenceph. Clin. Neurophysiol., Vol. 53, pp. 405-416.

Friedlander, B. 1982 a. System identification techniques for adaptive noise cancelling. IEEE Trans. ASSP, Vol. ASSP-30, No. 5, pp. 699-709.

Friedlander, B. 1982 b. Lattice filters for adaptive processing. Proc. IEEE, Vol. 70, No. 8, pp. 829-867.

Fu, K.S. 1982. Syntactic pattern recognition and applications, New Jersey, Prentice-Hall.

Gardiner, T., McWhirter, J.G. and Shepherd, T.J. 1985. Noise cancellation studies using a least-squares lattice filter. Proc. ICASSP, Tampa, pp. 1173-1176.

Gardner, W.A. 1981 a. Two-stage adaptive noise cancellation for slowly fluctuating signals. Proc. IEEE, Vol. 69, No. 4, 487.

Gardner, W.A. 1981 b. Cascade configurations for recursive-like adaptive noise cancellation. Proc. IEEE, Vol. 69, No. 7, pp. 846-847.

Geddes, L.A. and Baker, L.E. 1975. Principles of applied biomedical instrumentation, Toronto, Wiley.

Gibson, C.J. 1982. Adaptive lattice filtering for radar applications. CRL Internal Reports, McMaster University, Hamilton, No. CRL-97.

Goodwin, G.C. and Payne, R.L. 1977. Dynamic system identification, New York, Academic Press.

Goodwin, G.C. and Sin, K.S. 1984. Adaptive filtering prediction and control, New Jersey, Prentice-Hall.

Gritton, C.W.K. and Lin, D.W. 1984. Echo cancellation algorithms. IEEE ASSP Magazine, Vol. 1, No. 2, pp. 30-38.

Haykin, S. 1986. Adaptive filter theory, New Jersy, Prentice-Hall.

Itakura, F. and Saito, S. 1971. Digital filtering techniques for speech analysis and synthesis. Proc. 7th Int. Cong. Acoust., Budapest, No. 25-C-1, pp. 261-264.

Jewett, D.L. and Romano, M.N. 1972. Neonatal development of auditory system potentials averaged from the scalp of rat and cat. Brain Research, Vol. 36, 101.

Kailath, T. 1981. Lectures on Wiener and Kalman filtering, New York, Springer-Verlag.

Kaveh, M., Bruzzone, S. and Torres, F. 1978. A new method for the estimation of average evoked potentials. IEEE Trans. SMC, Vol. SMC-8, No. 5, pp. 414-417.

Kiloh, L.G., McComas, A.J., Osselton, J.W. and Upton, A.R.M. 1981. Clinical Electroencephalography, London, Butterworths.

Lam, C., Zimmerman, K., Simpson, R.K., Katz, S. and Blackburn, J.G. 1982. Classification of somatic evoked potentials through maximum entropy spectral analysis. Electroenceph. Clin. Neurophysiol., Vol. 53, pp. 491-500.

Larson, H. and Lai, D.C. 1980. Walsh spectral estimates with applications to the classification of EEG signals. IEEE Trans. BME, Vol. BME-27, No. 9, pp. 485-492.

Lee, D.T.L. 1980. Canonical ladder form realizations and fast estmation algorithms, Ph.D. Dissertation, Department of Electrical Engineering, Stanford University, Stanford, CA.

Lee, D.T.L., Morf, M. and Friedlander, B. 1981. Recursive least squares ladder estimation algorithms. IEEE Trans. ASSP, Vol. ASSP-29, No. 3, pp. 627-641.

Ljung, L. and Soderstrom, T. 1983. Theory and Practice of recursive identification. Massachusetts, MIT Press.

Ljung, S. and Ljung, L. 1985. Error propagation properties of recursive least-squares adaptation algorithms. Automatica, Vol. 21, No. 2, pp. 157-167.

Madhavan, G.P. 1980. Theory of current source density analysis. Private communication. Dept. of Neurobiology, NEOUCOM, Kent, USA.

Madhavan, G.P., de Bruin, H., Upton, A.R.M. and Jernigan, M.E. 1983. Syntactic pattern recognition of brain stem auditory evoked potentials. Proc. IEEE Int. Conf. SMC, Bombay, pp. 637-640.

Madhavan, G.P., de Bruin, H. and Upton, A.R.M. 1984. Evoked potential processing and pattern recognition. Proc. 6th Annual Conf. IEEE EMBS, Los Angeles, pp. 699-702.

Madhavan, G.P., de Bruin, H., Upton, A.R.M. and Jernigan, M.E. 1986. Classification of brain stem auditory evoked potentials by syntactic methods. Electroenceph. Clin. Neurophysiol., accepted for publication.

Makhoul, J. 1975 a. Stable and efficient lattice methods for linear prediction. IEEE Trans. ASSP, Vol. ASSP-25, pp. 423-428.

Makhoul, J. 1975 b. Linear prediction: a tutorial review. Proc. IEEE, Vol. 63, No. 4, pp. 561-580.

McClellan, J.H., Parks, T.W. and Rabiner, L.R. 1973. A computer program for designing optimum FIR linear phase digital filters. IEEE Trans. AU, Vol. AU-21, pp. 506-526.

Morf, M. 1977. Ladder forms in estimation and system identification. Proc. 11th Annual Asilomar Conf. on CSC, Pacific Grove, pp. 424-429.

Nohara, T. 1985. Syntactic pattern recognition of visual evoked potentials. Internal Report, Dept. Biomedical Engineering, McMaster University, Hamilton.

Ono, K., Baba, H., Mori, K., Nakatsuka, K. and Chiba, G. 1981. A new pattern discriminant method for evoked potentials. Intern. J. Neruoscience, Vol. 14, pp. 1-5.

Pack, J.D. and Satorius, E.H. 1979. Least squares, adaptive lattice algorithms. Technical Report Naval Ocean Systems Center, San Diego, No. 423.

Puthenpura, S., Sinha, N.K. and Vidal, O.P. 1986. Application of M-estimation in robust recursive system identification. IFAC Symp. Stochastic Control, Vilnius, U.S.S.R., accepted for publication.

Rauner, H., Wolf, W. and Appel, U. 1983. New perspectives to noise reduction in evoked potential processing. Signal processing: theories and applications, Schussler, H.W. (ed.), Amsterdam, Elsevier Science Publishers.

Reddy, V.U., Egardt, B. and Kailath, T. 1981. Optimized lattice-form adaptive line enhancer for a sinusoidal signal in broad-band noise. IEEE Trans. ASSP, Vol. ASSP-29, No. 3, pp. 702-709.

Robinson, E.A. and Treitel, S. 1980. Maximum entropy and the relationship of the partial autocorrelation to the reflection coefficients of a layered system. IEEE Trans. ASSP, Vol. ASSP-28, No. 2, pp. 224-235.

Rodriguez, M., Williams, R. and Carlow, T. 1981. Signal delay and waveform estimation using unwrapped phase averaging. IEEE Trans. ASSP, Vol. ASSP-29, No. 3, pp. 508-513.

Samson, C. 1982. A unified treatment of fast algorithms for identification. int. J. Control, Vol. 35, No. 5, pp. 909-934.

Samson, C.G. and Reddy, V.U. 1983. Fixed point error analysis of the normalized ladder algorithm. IEEE Trans. ASSP, Vol. ASSP-31, No. 5, pp. 1177-1191.

Satorius, E.H. and Pack, J.D. 1982. Application of least squares lattice algorithms to adaptive equalization. IEEE Trans. COM, Vol. COM-29, No. 2, pp. 136-142.

Shensa, M.J. 1981. Recursive least squares lattice algorithms - a geometrical approach. IEEE Trams. AC, Vol. AC-26, No. 3, pp. 695-702.

Shichor, E. 1982. Fast recursive estimation using the lattice structure. Bell System Technical Journal, Vol. 61, No. 1, pp. 97-115.

Silvia, M.T. and Robinson, E.A. 1979. Deconvolution of geophysical time series in the exploration for oil and natural gas, Amsterdam, Elsevier Scientific Publishing Company.

Sinha, N.K., Law, S.Y. and Mamen, R. 1979. Microcomputer-based on-line state estimation with applications to satellites. AGARD Conf. Proc., Ottawa, No. 272, 17-1.

Sinha, N.K., Law, S.S.Y. and Li, M.H. 1981. Microcomputer-based observer for nonlinear systems. IEEE Trans. IECI, Vol. IECI-28, No. 2, pp. 136-140.

Sinha, N.K. and Kuszta, B. 1983. Modeling and identification of dynamic systems, New York, Van Nostrand Reinhold.

Stockman, G., Kanal, L. and Kyle, M.C. 1976. Structural pattern recognition of carotid pulse waves using a general waveform parsing system. Comm. ACM, Vol. 19, pp. 688-695.

Swanson, D.C. and Symons, F.W. 1984. Sources of numerical errors in both the square-root normalized and unnormalized least squares lattice algorithms. Proc. ICASSP, San Diego, No. 45-4.

Toda, M. 1970. Waves in nonlinear lattice. Suppl. Progr. Theoretic. Phys., No. 45, pp. 174-200.

Tou, J.T. and Gonzalez, R.C. 1974. Pattern recognition principles, Massachussets, Addison-Wesley.

Walter, D.O. 1969. A posteriori "Wiener filtering" of average evoked responses. Electroenceph. Clin. Neurophysiol., Vol. 43, 476.

de Weerd, J.P.C. and Martens, W.L.J. 1978. Theory and practice of a posteriori "Wiener" filtering of average evoked potentials. Biol. Cybernetics, Vol. 30, pp. 81-94.

de Weerd, J.P.C. 1981. A posteriori time-varying filtering of averaged evoked potentials; introduction and conceptual basis. Biol. Cybernetics, Vol. 41, pp. 211-222.

de Weerd, J.P.C. and Kap, J.I. 1981. A posteriori time-varying filtering of averaged evoked potentials; mathematical and conceptual aspects. Biol. Cybernetics, Vol. 41, pp. 223-234.

de Weerd, J.P.C. 1984. Time-varying filtering: what it does and how it works. Nicolet Potentials.

Wiberg, D.M., Baskin, F. and Linsay, R.D. 1985. On the dynamics of recursive least-squares and lattices. IFAC Symp. Identification and System Parameter Estimation, York, U.K., pp. 879-883.

Widrow, B. and Hoff, M.E. 1960. Adaptive swithcing circuits. IRE WESCON Conv. Rec., Part 4, pp. 96-104.

Widrow, B., Glover, J.R., McCool, J.M., Kaunitz, J., Williams, C.S., Hearn, R.H., Zeidler, J.R., Dong, E. and Goodlin, R.C. 1975. Adaptive noise cancellling: principles and applications. Proc. IEEE. Vol. 63, No. 12, pp. 1692-1716.

Wiener, N. 1949. The extrapolation, interpolation and smoothing of stationary time series, with engineering applications. New York, Wiley.

Winski, R. and Allinson, N.M. 1984. Adaptive processing of brain evoked potentials. IEE Colloq. Adaptive Proc. & Biomed. Applications, London, U.K., 7-1.

Woody, C.D. 1967. Characterization of an adaptive filter for the analysis of variable latency neuroelectric signals. Med. & Biol. Eng., Vol. 5, pp. 539-553.

Yu, K. and McGillem, C.D. 1983. Optimum filters for estimating evoked potential waveform. IEEE Trans. BME, Vol. BME-30, No. 11, pp. 730-737.